AFIT/GOR/ENS/93M-25

# MODIFIED GOODNESS-OF-FIT TESTS FOR THE WEIBULL DISTRIBUTION

THESIS

Erol YÜCEL
First Lieutenant, TUAF

AFIT/GOR/ENS/93M-25

# THESIS APPROVAL

STUDENT: 1Lt Erol YÜCEL CLASS: GOR-93M

THESIS TITLE: MODIFIED GOODNESS-OF-FIT TESTS FOR THE WEIBULL DISTRIBUTION

DEFENSE DATE: March 8, 1993

| COMMITTEE: | NAME/DEPARTMENT | SIGNATURE |
|---|---|---|
| Advisor | Dr Albert H. Moore/EN | Albert H Moore |
| Reader | Dr Joseph P. Cain/ENS | Joseph P. Cain |

AFIT/GOR/ENS/93M-25

# MODIFIED GOODNESS-OF-FIT TESTS FOR THE WEIBULL DISTRIBUTION

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Operations Research

Erol YÜCEL, B.S.
First Lieutenant, TUAF

March, 1993

Approved for public release; distribution unlimited

## *Preface*

The purpose of this research is to provide a new goodness of fit test for the three parameter Weibull distribution.

I would like to express my thanks to all the AFIT faculty members, especially to Dr. Albert H.MOORE for his assistance and guidance in this research as my thesis advisor. I will always admire him as a person and be proud of being an Albert H.MOORE student. My thanks also go to Dr.J.CAIN as my reader.

I would like to thank to my best friends, Tamer and his wife Ozlem for supporting, encouraging me and not letting me forget the wonders of Turkish cuisine during my stay in the U.S.A.

Finally, My thanks go to my family, my father Ibrahim, my mother Ayse Hanim, my sister Selma, my brother Birol and all my relatives in Zafertepe Calkoy. Without their support this thesis effort would not be possible.

Above all, I promise to serve to my country and people with my best to be able to express my thanks for sending me to challenge two years in AFIT.

Erol YÜCEL

*Table of Contents*

## *List of Figures*

## List of Tables

AFIT/GOR/ENS/93M-25

## *Abstract*

This research will prod :ce a new modified Anderson–Darling and $\tilde{W}^*$ Goodness of fit statistics for the three parameter Weibull Distribution when all parameters unknown and estimated by Maximum Likelihood-Minimum Distance combination. The critical values for each statistic , sample size and $\alpha$ levels 0.20,0.15,0.10,0.05 and 0.01 are produced. The Monte Carlo Simulation used 5000 repetitons for sample sizes of 10,15,20,25,30,40, and 50 with the Weibull shape equals to 3, scale equals to 4 and location equals to 10. The power study is made for the same sizes as above with the hypothesized Weibull Distribution against 8 other distributions.

# MODIFIED GOODNESS-OF-FIT TESTS FOR THE WEIBULL DISTRIBUTION

## I. Introduction

### 1.1 Background

The Air Force spends billions of dollars every year for data from experimentation to solve the problems growing in size and complexity. Consequently, the optimal utilization of data in making decision and a careful design and analysis of the experiment become very important.

In order to solve complex problems, Military analysts frequently use Simulation, or statistical models rather than analytical techniques or mathematical formulation. They have to give special attention to choosing particular distributions using sample data to characterize random elements of the system under study. Because the quality of the analysis, decision or prediction depends on the appropriateness of the models used.

If a theoritical probability distribution has been fitted to some observed data and used as input to the simulation model, the adequacy of the fit can be assessed by the graphical plots and goodness-of-fit tests. In order to carry out a simulation using random inputs , we have to specify their probability distributions. Then given that the input random variables to a simulation model follow particular distributions, the simulation proceeds through time by generating random values from these distributions.

Our concern here is how the analyst might go about specifying these input probability distributions. Almost all real systems contain one or more sources of randomness. Furthermore, it is generally necessary to represent each source of system randomness i . probability distribution in the simulation model. The failure to choose the correct distribution can also affect of a model's results. The choice of probability distributions can evidently have a large impact on the simulation output and, potentially, on the quality of the decisions made with the simulation results.

### *1.2 Definitions*

1. A *distribution* is a single or multi-parameter theoretical, statistical model of data, often used to predict the behavior of a population of entities by studying a sample of it.

2. Given a random sample $X_1, \ldots, X_n$ drawn from a distribution with cumulative distribution function (cdf) F, then the *empirical distribution function (edf)* is defined as

$$F_n = (thenumberofX_i's \leq x)/n$$

. For all x values $F_n(x)$ converges for large samples to F(x), the value of the underlying distribution's cdf at x [25:8]. The graphical representation of EDF, CDF relationship is in figure(1.1).

3. A *statistic* is any function of the random variables constituting one or more samples, provided that the function does not depend on any unknown parameter values [6:231].

4. *Goodness-of-fit tests* measure from the observed data the ability of the particular statistical distribution to model the underlying random variable. The most commonly used goodness-of-fit tests are the Chi-square, the Kolmogorov-Smirnov (K-S), the Cramer-von Mises (C-vM), and the Anderson-Darling (A-D).

   Before applying any goodness-of-fit test, the researcher must complete four steps to determine which distribution is suggested the data. These are

   - collecting data for the study problem,

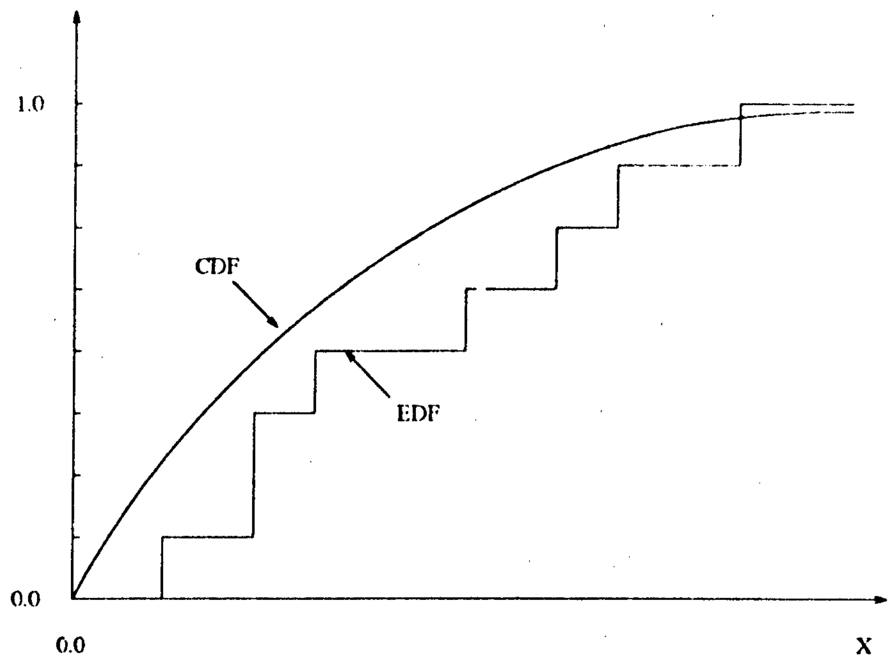


Figure 1.1. This figure describes the CDF and EDF graphically.

- selecting which statistical distribution best characterizes the data,
- estimating parameters of the suggested distribution,
- determining if the data follows the chosen statistical distribution as selected using one of the above goodness-of-fit tests which has the highest power.

5. A *statistical hypothesis*, or hypothesis, is a claim either about the value of a single population characteristic or about the values of several population characteristics.

   In any hypothesis testing problem, there are two contradictory hypotheses under consideration. The objective is to decide, based on sample information, which of the hypotheses is correct. The claim initially believed to be true is called the *null hypothesis* and denoted by $H_o$. The other claim in a hypothesis testing problem is called the *alternative hypthesis* and is denoted by $H_a$. Thus we might test $H_o$: $\mu = 0.75$ against the alternative $H_a : \mu \neq 0.75$. Only if sample data strongly suggests that $\mu$ is something other than 0.75 should the null hypothesis be rejected. In the absence of such evidence, $H_o$ should not be rejected, since it is still quite plausible [6:283-284].

6. A *test procedure* is a rule, based on sample data, for deciding whether to reject $H_o$. This procedure has two constituents:

   a *test statistic*, a function of the sample data on which the decision (reject $H_o$ or do not reject $H_o$) is to be based, and

   a *rejection region*, the set of all test statistic values for which $H_o$ will be rejected. The null hypothesis will then be rejected if and only if the observed or computed test statistic value falls in the rejection region [6:284-285].

7. *Power* is the probability of rejecting the hypothesis when in fact is false. The higher the power of a test, the lower the chance of accepting a distribution when it is false. If the test rejects the hypothesis, one must return to the second step selecting and testing another

8. An *estimator* is a numerical function of the data. There are many ways to specify the form of an estimator for a particular parameter of a given distribution, and many alternative ways to evaluate the quality of an estimator [16:368].

   distribution.

### *1.3 Scope*

In this research I will study extensively three parameter Weibull distribution with unknown parameters. Since I will estimate the parameters, I need to build tables for each statistic, density, estimator and empirical distribution function. The scope of the analysis will be to adjust the range of the tables in conjunction with the time and computer resources available.

### 1.4 *Problem Statement*

There are several goodness-of-fit tests for the two parameter Weibull distribution with shape and scale parameters. However the two parameter Weibull distribution assumes the smallest possible random variable is zero. When this assumption does not hold we have to add another parameter ( location parameter ) to adjust the smallest possible value to zero. In this case we can use the three parameter weibull distribution with scale, shape and location parameter. But there is very little known about goodness-of-fit tests for the three parameter Weibull distribution when all parameters are unknown.

### 1.5 *Research Objective*

The purpose of this study is to derive critical values for a new goodness-of-fit test statistic and to examine the power of the new test against the power of alternative tests. Alternative tests may be derived for the comparison.

### 1.6 *Summary*

Most of the books about statistics do not include enough information how to choose distributions to model the system behavior and how to test the distributions chosen. This kind of lack of information lead the statistical practitioners to choose distributions well known and easy to apply, but not representing the data .

Most of the modelling work is highly dependent on the distributions chosen to represent the random elements in the system modelled. If one does not have enough information about how to choose distributions or does not test the distribution chosen then the study is subject to incorrect results.

This study of statistical data will help the Air Force to better predict the Reliability and maintainability of systems. Because in the litterature and in the real life when using Weibull distribution as a model analysts consider minimum life of product as zero. But there are cases where the minimum life is not zero. Most of the researchers did not adress this problem. Because when all the parameters known, but shape parameter, null distribution theory will depend on the true values of the parameters estimated. But when the location and scale parameters are unknown and estimated by appropriate methods, the distributions of EDF statistics will not depend on the true values of the estimated parameters [25:103].

### *1.7 Support Requirements*

This research will require AFIT computer resources. A program will be written in Sun Pascal 2. For educational purposes and validation of the Pascal codes MCAD will be used.

# II. Literature Review

## 2.1 Background

This thesis effort is focused on the Weibull distribution with three parameters unknown. In this research Maximum Likelihood Estimation (MLE) technique and Minimum Distance Estimation (MDE) will be used to obtain the point estimates of the unknown parameters of the Weibull distribution. In the literature review, special attention is given to parameter estimation techniques, goodness-of-fit test statistics, random number generation techniques and generators.

The three parameter Weibull distribution is applicable to many random phenomenon [11:164]. It has been found to provide a reasonable model for lifetimes of many type of unit, such as vacuum tubes, ball bearings and composite materials [4:17], for time to complete some task [16:333], and for interarrival and service times ( actually, the exponential distribution is a special case of both the gamma and the Weibull distributions) [1:132]. Especially, in reliability estimation the Weibull distribution is second in use after the exponential ( Unfortunately in many cases, it is used because it is easy to apply rather than because it is a choice based on a through understanding of the fundamentals [15:233].

The following explanation shows intuitively that sometimes the Weibull distribution provides a better model than the exponential distribution does.

The tail of the Weibull distribution may decline more rapidly or less rapidly than that of exponential distribution. In practice, this means that if there are more large service times than exponential can account for, a Weibull distribution may provide a better model of these service times [1:132-133].

Gallagher proved that the Weibull distribution allowed the pdf to fit data that was actually from the gamma distribution and tested the robustness of the Weibull with respect to other probability distributions.

## 2.2 *Maximum Likelihood*

In the Hypothesis testing, usually parameters are unknown, and must be estimated from the observed data.

In this research, MLE is choosen for the following reasons :[16:350-354]

- MLEs have several desirable properties often not enjoyed by alternative methods of estimation, e.g., least-squares estimation, unbiased estimators, Modified Moment Estimators and the method of moments; As Cohen [3:31] noted although calculation of Moment estimators (ME) requires considerably less computational effort than MLE, it should be remembered that estimate variances of the MLE are smaller than corresponding variances of the ME. However, ME are applicable over the entire parameter space, whereas computational problems arise with MLE when $\beta < 1$.

  1. For the most common distributions, the MLE is unique; that is, L($\hat{\theta}$) is *strictly* greater than L($\theta$) for any other value of teta.
  2. Although MLEs need not be biased, in general, the asymptotic distribution (as $n \to \infty$ ) of $\hat{\theta}$ has mean equal to $\theta$.
  3. MLEs are *invariant*; that is, if fa=h($\theta$) for some function h, then the MLE of fa is h($\hat{\theta}$). (unbiasedness is not invariance) For example, the variance of an exponential(beta) random variable is $\beta^2$, so the MLE of this variance is $\bar{x}_n^2$.
  4. MLEs are asymptotically normally distributed;
  5. MLEs are *strongly consistent*; that is, $\lim_{n\to\infty} \hat{\theta} = \theta$

- The use of MLEs turns out to be important in justifying the chi-square goodness-of-test;
- The central idea of ML estimation has a strong intuitive appeal.

But finding MLEs for the three parameter Weibull distribution is very difficult.

The coefficient of variation $cv = \sqrt{S_n^2/[\overline{x}_n - \gamma]^2}$, where $\gamma$ is known location, can sometimes provide useful information about the form of a continuous distribution. For the Weibull distributions, cv is greater than, equal to, or less than 1 when the shape parameter is less than, or equal to, or grater than 1, respectively. This summary statistics is not particularly useful for other distributions, except Gamma, Exponential, Lognormal [16:358].

It is possible to use the Kurtosis, which is a measure of the tail weight of a distribution, as a function of distribution parameters. However, Law and Kelton did not found the Kurtosis to be very useful for disciminating among distributions.

Some of the distributions have range $[0,\infty)$ ( such as gamma, Weibull, lognormal, exponential ). Thus, if a random variable X has any of these distributions, in practice sometimes X cannot be less than some positive value $\gamma$ ( such as service time ). In this kind of situation, if range $[0,\infty)$ is used, even though $P(X < \gamma)=0$, there is a chance to generate a random variable less than $\gamma$.

To solve this kind of problem, we can shift the distribution $\gamma$ distance to the right. Here $\gamma$ is called as location parameter. Then the range of the shifted distribution becomes $[\gamma,\infty)$. But the shifted Weibull, gamma (global) MLEs are not defined very well [2]. That is, the likelihood function L can be made infinite by choosing $\hat{\gamma} = X_{(1)}$( the smallest observation in the sample), which results in admissible values for the other parameters [16:401]. The same authors criticized the approach suggested by Harter and Moore seeking a local, as opposed to global, maximum point of L [10]. But this approach is very simple in concept. But as Harter and Moore pointed out, when location estimate is bigger than $X_{(1)}$ numerical problems occur because $\ln(X_{(1)} - \gamma)$ does not exist. Harter and Moore suggested censoring the random variables less than or equal to $\gamma$, then continuing the estimation of the the parameters left.

Cheng and Amin [2] proposed an alternative estimation method for three parameters, called *maximum product of spacing (MPS) estimation*. This method can be used when MLE fails. This method solves three equations in three unknowns using a numerical approach.

Dubey [7] suggested another method for three parameter estimation problem. In practice this method first estimate the location parameter $\gamma$ by

$$\tilde{\gamma} = \frac{X_{(1)}X_{(n)} - X_{(k)}^2}{X_{(1)} + X_{(n)} - 2X_{(k)}} \tag{2.1}$$

where $k$ is the smallest integer in 2,3,...,$n-1$ such that $X_{(k)} > X_{(1)}$. It is shown by Dubey [7] that $\tilde{\gamma} < X_{(1)}$ if and only if $X_{(k)} < \left[\frac{X_{(1)}+X_{(n)}}{2}\right]$ . Zanakis [29] in his research concluded that $\tilde{\gamma}$ was accurate for the Weibull distribution. Given $\tilde{\gamma}$ as location parameter, two parameter MLE can be applied for the shape and scale parameter after subtracting out $\tilde{\gamma}$ for all the observations.

Also, Johnson [14] discusses some other alternative estimators based on Order Statistics. He also noted that the MLEs are regular ( in the sense of having the usual asymptotic distribution ) only for shape estimate> 2. If it is known that 0 < Shape Estimate < 1, then $min(X_1, ..., X_n)$ is a super efficient estimator for the location parameter. [14:256].

Usually location parameter is assumed zero. But a value of location less than zero could indicate failure in storage [13:4-47]. Hirose [12:310] discuss the location parameter in his paper as follows : In failure analysis (especially in electrical engineering) it is well known that failures follow the Weibull cdf and there seems to exist certain point, greater than zero, in the Weibull cdf under which a breakdown will not occur, or at least will be very rare. Since very low failure probabilities are expected in power electric equipment, electrical engineers consider it crucial to estimate this point.

Hirose [12:330] proposed an algorithm of MLE comprised of three parts.

- Determining appropriate initial values for Newton-Raphson method.
- Finding the approximate values by using the line search algorithm.
- Solving the three simultaneous likelihood equations by Newton-Raphson method.

He concluded that the larger the shape value, the more often the parameters fail to converge in MLE. But using Harter and Moore's algorithm this problem was not encountered.

A scale parameter determines the scale ( or unit ) of measurement of the values in the range of the distribution. A change in scale parameter compresses or expands the associated distribution without altering its basic form. A change in shape alters a distribution's properties (e.g., skewness) more fundementally than a change in location or scale.

It is rare to know the parameters of a distribution being tested. In this research, I will use two different estimation techniques: Maximum Likelihood Estimation (MLE) and Minimum Distance Estimation (MDE).

The MLE selects as distribution parameters whose values that maximizes the likelihood of the observed sample, where the likelihood function is the joint density function. Therefore, the probability of the observed sample is maximized by the choice of the distribution parameter values.

In recent studies by Dr. A.H.MOORE and his students , maximum likelihood estimates with minimum distance estimation of location performed very well. Mark GALLAGHER showed that estimating location by minimizing AD statistics given Maximum likelihood estimates was the best method among several alternatives including MLE. But in his study, he did not let location parameter go to below 0. Also, in the location procedure he had a bug which did not give correct estimates of some samples. This was corrected and the whole program was rerun only for WEIBULL tables. Surprisingly, minimizing the AD statistics was better than other techniques he investigated, approximately 900 times out of 1000 repetitions.

## 2.3 Random Number Generator

Any one who considers arithmetical methods of producing random digits is, of course, in a state of sin. For as has been pointed out several times, there is no such thing as a random number–there are only methods to produce random numbers, and a strict arithmetic procedure of course is not such a method....We are here dealing with mere "cooking recipes" for making digits.... [27].

Arithmetically generated random numbers ( usually called *pseudorandom*) can be defined as numbers that appear independently drawn from the U(0,1).

The methodology of generating random numbers from a distribution is first to obtain random variates from the uniform distribution on the interval [0,1], then transforming these IID random numbers in a way determined by the distribution.

A good arithmetic random number generator should possess several properties:

- Above all, the numbers produced should appear to be distributed uniform- ly on [0,1] and should not explicit any correlation with each other; otherwise, the simulation's results may be completely invalid.
- From a practical standpoint, we would naturally like the generator to be fast and avoid the need for a lot of storage.
- We would like to be able to reproduce a given stream of random numbers exactly, for at least two reasons. First, this can sometimes make debugging or verification of the computer program easier.
- There should be provision in the generator for producing several separate "streams" of random numbers. A stream is simply a subsegment of the numbers produced by the generator, with one stream beginning where the previous stream ends. We can think of the different streams as being separate and independent generators. Thus, the user can "dedicate", a particular stream to a particular source of randomness in the simulation.

In the history of random number generation, throwing dice, drawing numbers from a urn, or dealing out cards has received a remarkable attention by statisticians. Later, mechanized devices and electronic-random number generators have been used to generate random numbers more efficiently and quickly. But as modern large-scale simulations become possible by use of computers, methods to generate random numbers by arithmetic ways has become neccesary. Arithmetic methods use a fixed mathematical formula to generate random numbers. This kind of first generator proposed by von Neumann and Metropolis [27] called *midsquare method.* But this method failed because for some starting values it generates 0 quickly and stays there forever. Later in 1951, Lehmer [17]introduced *linear congruential generators* (LCGs). This generator uses a recursive formula

$$Z_i = (aZ_{i-1} + c) \bmod m) \tag{2.2}$$

where $m$ (the *modulus*), $a$ (the *multiplier*), $c$ (the *increment*), and $Z_0$ (the *seed* ) are all nonnegative integers satisfying the following constraints : $0 < m$, $a < m$, $c < m$, and $Z_0 < m$. From equation(3), $0 \leq Z_i \leq (m - 1)$. By dividing $Z_i$ by m one can get pseudo random numbers on U(0,1). LCGs has a looping behavior ( the same sequence of random numbers will repeat itself whenever $Z_i$ is equal to the $Z_0$. This lenght of cycle is called the *period* of a generator. This period is at most m. When the period is m, it is called full period and any starting value will produce a $m - 1$ different pseudorandom numbers. But if the period is less than m, than the period will depend on only the starting value. Full period LCGs are desirable but some of them can show nonuniformality leaving big gaps in the sequence of possible values. Therefore m, a, and c parameters should be chosen very carefully. The following theorem, as given in Law and Kelton [16] can be helpful in choosing these parameters.

The LCG defined in Eq.(1) has full period if and only if the following three conditions hold:

- the only positive integer that (exactly divides both m and c is 1.
- if $q$ is a prime number that divides $m$, then $q$ divides $a - 1$
- if 4 divides $m$, then 4 divides $m - 1$

When $c > 0$ LCGs are called *mixed*, otherwise ($c = 0$) they are called *multiplicative* LCGs. In this research, multiplicative LCG will be used. GALLAGHER and CROWN used different LCGs. They did not show why and how they choose their LCG parameters (a, c, and m). I tried both sets of parameters. When Crown's parameters are used in Sun Pascal 2., Integer overflow experimented resulting negative pseudorandom numbers on U(0,1). But this did not occur in CSC pascal. Thus, One should be very careful choosing LCG parameters . For this reason $m$ is chosen as $2^{31} - 1$ (which is a prime) and $c$ is chosen as 16807 which is used in IMSL routines because it has the fastest execution time among three possible c values offered in IMSL generators. Also Some of the simulation languages uses 16807 as multiplier [16:357]. Later, using the Chi square test as shown in Law and Carson [16:437]. 5000 pseudorandom numbers are tested in MCAD. The chi-square test with all parameters are known is used to check whether the pseudorandom numbers generated by using this generator appear to be uniformly distributed between 0 and 1. We divide [0,1] into $k$ subintervals of equal length and generate $U_1, U_2, ..., U_n$.For $j = 1, 2, ..., k$, let $f_i$ be the number of the $U_i$'s that are in the $j$th subinterval, and let

$$\chi = \frac{k}{n} \sum_{i=1}^{k} (f_i - \frac{n}{k})^2$$

(2.3)

Then for large $n$, $\chi^2$ will have an approximate chi-square distribution with $k - 1$ df under the null hypothesis that the $U_i$'s are independently identically distributed (IID) U(0,1) random variables. Thus we reject this hypothesis at level $\alpha$ if $\chi^2 > \chi^2_{k-1,1-\alpha}$,where $\chi^2_{k-1,1-\alpha}$ is the upper $1 - \alpha$ critical point of the chi square distribution with $k-1$ degrees of freedom (df). For the large values of $k$ , the following approximation can be used ;

$$\chi^2_{k-1,1-\alpha} \approx (k-1)\left(1-\frac{2}{9(k-1)}+z_{1-\alpha}\sqrt{\left[\frac{2}{9(k-1)}\right]^3}\right) \tag{2.4}$$

where $z_{1-\alpha}$ is the upper $1-\alpha$ critical point of the Normal(0,1) distribution.

### *2.4 Random Variate Generation Techniques*

In this section, The most widely used techniques for generating random variates will be briefly explained, such as inverse transform technique, the convolution method, and acceptance-rejection technique.

*2.4.1 Inverse transform technique.* This technique is very straighforward. It can be used when the inverse of cdf $F(x)$ has an explicit formula. For example since the Gamma distribution does not have an explicit cdf $F(x)$ , this method can not be used to generate the random gamma deviates. A step-by-step procedure for the inverse transform technique, illustrated by the exponential distribution, is as follows [1:294].

- Compute the cdf of the desired random variable $X$. For the exponential distribution, the cdf is $F(x) = 1 - \exp^{-\lambda x}, x \geq 0$.
- Set $F(x) = R$ on the range of $X$. R has a uniform distribution over the interval (0,1). For the exponential distribution, it becomes $1 - \exp^{-\lambda x}$, $x \geq 0$.
- Solve the equation $F(x) = R$ for $X$ in terms of R. For the exponential distribution, the solution proceeds as follows:

  1. $1 - \exp^{-\lambda x} = R$
  2. $\exp^{-\lambda x} = 1 - R$
  3. $-\lambda X = \ln 1 - R$

4. $X = \frac{-1}{\lambda}\ln(1-R)$ This equation is called a random variate generator for the exponential distribution. In general this equation is written as $X = F_R^{-1}$. Generating a sequence of values is accomplished through next step.

5. Generate (as needed) uniform random numbers $R_1, R_2, R_3, \ldots$ and compute the desired random deviates by

$$X_i = F^{-1}(R_i) \tag{2.5}$$

For the exponential case,

$$X_i = \frac{-1}{\lambda}\ln(1-R_i) \tag{2.6}$$

One simplification to this equation is to replace $(1-R_i)$ by $R_i$.

The uniform , Weibull random generators ( using the inverse transform technique) are as follows:

- Uniform random generator : $X = a + (b-a)R$, given $a \le X \le b$.
- Weibull random generator : $X = \alpha[-\ln 1 - R]^{\frac{1}{\beta}} + \gamma$, given $x \ge 0$

*2.4.2 Convolution Method.* The probability distribution of a sum of two or more independent random variables is called a convolution of the distributions of the original variables. The convolution method thus refers to adding together two or more random variables to obtain a new random variable with the desired distribution. This technique can be applied to obtain Erlang variates, approximately normally distributed variates, and binomial variates. What is important is not the cdf of the desired random variable, but rather its relation to other more easily generated variates [1:317].

| i | 1 | 2 | 3 |
|---|---|---|---|
| $R_i$ | 0.1306 | 0.6597 | 0.7696 |
| $X_i$ | 0.1400 | 1.078 | [illegible] |

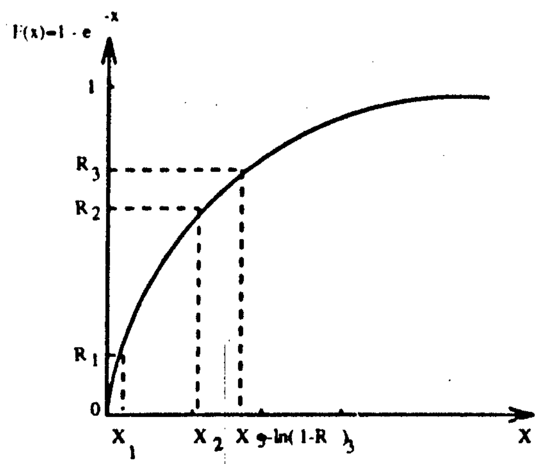


Figure 2.1. Inverse Transformation Technique

In this research, the Gamma and normal random deviates are generated by using this method as follows:

- An Erlang variable X with parameters $(K, \theta)$ is the sum of K independent exponential random variables, $X_i (i = 1, ..., K)$, each having mean $1/K\theta$; that is, $X = \sum_{i=1}^{K} X_i$. Since each $X_i$ can be generated by equation $X_i = \frac{-1}{\lambda} ln R_i$ with $1/\lambda = 1/K\theta$, an Erlang variate can be generated by

$$X = \sum_{i=1}^{K} \frac{-1}{K\theta} ln R_i$$

- if the Gamma distribution's shape parameter is integer, it is called Erlang distribution.

*2.4.3 Direct Transformation for the Normal Distribution.* Consider two standard normal random variables, $Z_1$,$Z_2$, plotted as a point in the plane as shown in Figure2 and represented in ploar coordinates as $Z_1 = B \cos\theta$ and

$Z_2 = B \sin\theta$. It is known that $B^2 = Z_1^2 + Z_2^2$ has the chi-square distribution with 2 degrees of freedom, which is equivalent to an exponential distribution with mean 2. Thus, the radius, B, can be generated by $B = (-2 \ln R)^{1/2}$. By the symmetry of the normal distribution, it seems reasonable to suppose, and indeed it is in this case, that the angle $\theta$ is uniformly distributed between 0 and $2\pi$ radians. In addition, the radius, $B$, and the angle, $\theta$, are mutually independent. Combining Equations (1) and (2) gives a direct method for generating two independent standard normal variates, $Z_1$ and $Z_2$, from two independent random numbers $R_1$ and $R_2$:

$Z_1 = (-2 \ln R_1)^{1/2} \cos 2\pi R_2$ and $Z_2 = (-2 \ln R_1)^{1/2} \sin 2\pi R_2$.

### *2.5 Summary*

In this section, Maximum Likelihood and Minimum Distance estimation techniques, random numbers and generators are discussed. Solutions proposed for some of the MLE problems are presented.

# III. Methodology

## 3.1 Introduction

In this section, Weibull distribution will be discussed.

### 3.1.1 Weibull Cumulative Distribution Function (CDF).

$$F(x;\theta,\beta,\delta) = 1 - e^{-\left(\frac{x-\delta}{\theta}\right)^{\beta}} \qquad (3.1)$$

where $\theta > 0$ is the scale parameter, $\beta > 0$ is the shape parameter, $\delta \geq 0$ is the location parameter

## 3.2 Weibull Probability Density Function (PDF)

$$f(x;\theta,\beta) = \frac{\beta}{\theta}\left(\frac{x}{\theta}\right)^{\beta-1} e^{-\left(\frac{x}{\theta}\right)^{\beta}} \qquad (3.2)$$

## 3.3 Weibull Distribution Properties

- The exponential($\beta$) and Weibull(1,$\beta$) distributions are the same.
- X $\sim$ Weibull($\alpha,\beta$) if and only if $x^{\alpha} \sim \exp(\beta^{\alpha})$
- The natural logaritm of a Weibull random variable has a distribution known as the *Extreme-Value* or *Gumbel distribution.*
- The Weibull(2,beta) distribution is also called a *Rayleigh distribution* with parameter $\beta$, denoted Rayleigh($\beta$). If Y and Z are independent normal random variables with mean 0 and variance $\beta^2$, then $X = (Y^2 + Z^2)^{1/2} \sim$ Rayleigh($2^{1/2}\beta$)

1. As $\alpha \to \infty$, the Weibull distribution becomes degenerate at $\beta$. Thus, Weibull densities for large alfa have a sharp peak at the mode.

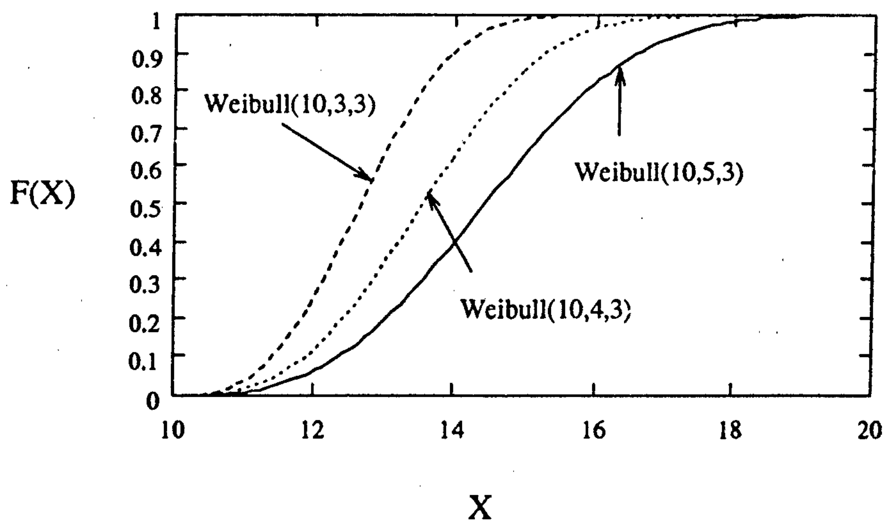


Figure 3.1. This figure shows the effect of a change in scale when shape and location are constant

2.

$$\lim_{x\to\infty} f(x) = \begin{cases} \infty & \text{if } \alpha < 1 \\ \frac{1}{\beta} & \text{if } \alpha = 1 \\ 0 & \text{if } \alpha > 1 \end{cases} \tag{3.3}$$

Following to figures do represent that the shape parameter is the key element in the Weibull distribution. As seen from figure one, when shape is constant, scale only streches or expands the CDF. But the following figure shows that the different shapes causes the CDF shape change.

### *3.4 Maximum Likelihood Estimators*

The method of maximum likelihood was first introduced by R.A.Fisher, a geneticist and statistician, in the 1920s. Most statisticians recommend this method, at least when the sample size is large, since the resulting estimators

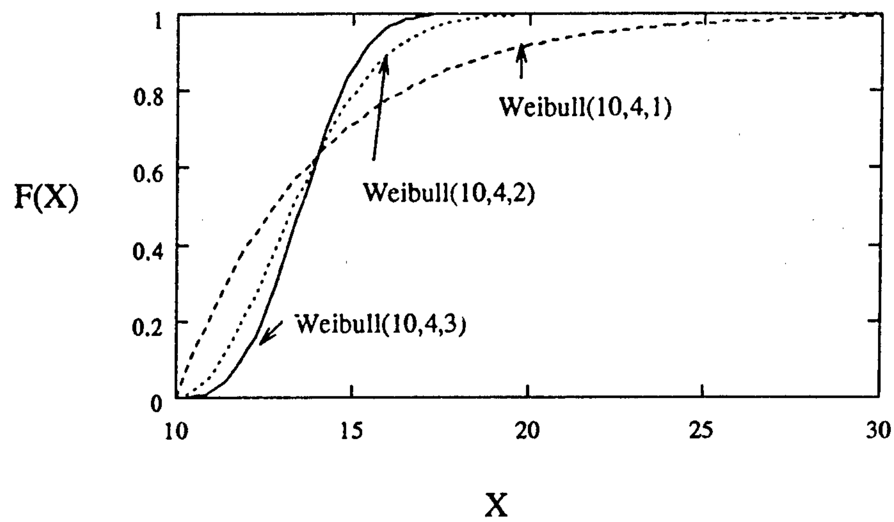


Figure 3.2. This figure shows the effect of a change in shape when scale and location are constant

have certain desirable efficiency properties. The likelihood function tells us how likely the observed sample is as a function of the possible parameter values. Maximizing the likelihood gives the parameter values for which the observed sample is most likely to have been generated, that is, the parameter values that "agree most closely" with the observed data.[6:247-248] But Some statisticians do not recommend to employ the MLE for the three-parameter Weibull distribution unless there is reason to expect that $\beta > 2.2$.[3:27]

The joint probability density function for a complete ordered random sample $X_i$,i=1,2...,n, $from the Weibull distribution is in Eq(4) L = (x_1, ..., x_n; \gamma, \theta, \beta) = \prod_{i=1}^{n} f(x_i; \gamma, \theta, \beta)(3.4)$

$$L = (\beta\theta^{-\beta})^n \prod_{i=1}^{n} (x_i - \gamma)^{\beta-1} e^{-\sum_{i=1}^{n}\left(\frac{x_i-\gamma}{\theta}\right)^{\beta}} \tag{3.5}$$

When $\beta < 1$, the distribution is reverse J-shaped and the likelihood function becomes infinite as $\gamma \Rightarrow x_1$, the smallest sample observation. Accordingly, in this situation the MLE of $\gamma$ would be $X_1$, but estimates of $\beta$ and $\theta$ would not exist. The Weibull distribution is bell-shaped when $\beta > 1$, and MLE in that case be found by simultaneously solving the system of equations obtained by equating to zero the partial derivativatives of the loglikelihood function with respect to the parameters. Taking the logarithm of L simplifies taking the derivative of the equation by converting the product of density function into summation. Taking the logarithm of L, differentiating, and equating partial derivatives to zero, we obtain

$$\frac{\partial \ln L}{\partial \gamma} = \beta\theta^{-\beta}\sum_{i=1}^{n}(x_i-\gamma)^{\beta-1} - (\beta-1)\sum_{i=1}^{n}(x_i-\gamma)^{-1} = 0 \tag{3.6}$$

$$\frac{\partial \ln L}{\partial \theta} = -n + \theta^{-\beta}\sum_{i=1}^{n}(x_i-\gamma)^{\beta} = 0 \tag{3.7}$$

$$\frac{\partial \ln L}{\partial \beta} = \frac{n}{\beta} + \sum_{i=1}^{n}\ln(x_i-\gamma) - \theta^{\beta}\sum_{i=1}^{n}(x_i-\gamma)^{\beta}\ln(x_i-\gamma) = 0 \tag{3.8}$$

These three equations can not be solved explicitly. But as Cohen showed $\theta$ can be eliminated from the last two equations to give

$$\left(\frac{\sum_{i=1}^{n}(x_i-\gamma)^{\beta}\ln(x_i-\gamma)}{\sum_{i=1}^{n}(x_i-\gamma)^{\beta}} - \frac{1}{\beta}\right) - \frac{1}{n}\sum_{i=1}^{n}\ln(x_i-\gamma) = 0 \tag{3.9}$$

Subsequently, $\hat{\theta}$ can be stated as

$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n}(x_i-\hat{\gamma})^{\frac{1}{\beta}}$

When we substitute this scale estimate into equation 3.5 , we get

$$(1-\beta)\sum_{i=1}^{n}(x_i-\gamma)^{-1} + n\beta\left(\frac{\sum_{i=1}^{n}(x_i-\gamma)^{\beta-1}}{\sum_{i=1}^{n}(x_i-\gamma)^{\beta}}\right) \tag{3.10}$$

But, still in order to get $\hat{\beta}$ and $\hat{\gamma}$ we need to solve the equations 4 and 5 iteratively. Actually when the location is known the first equation can be solved easily (but still iteratively) for $\hat{\beta}$. Cohen gives some techniques for a first approximation to $\hat{\beta}$ for use in iterative process, such as the Weibull coefficient of variation and its square as functions of the shape parameter. But since in this research the Harter and Moore algorithm used we really do not need an initial estimate for $\hat{\beta}$ . I tried several initial estimates for $\hat{\beta}$ I got the same estimates for all three parameters without having any problem. In their researchs, Miller, Gallagher and Crown, they censored the data when the location estimate is the first order statistic,$\hat{\gamma} = x_1$. Also they did not let the location parameter take negative values, once it did they let location parameter to 0. In this research, the location parameter allowed to go below zero. Also when the location parameter is bigger than first order statistic, the sample is thrown away.

### *3.5 Minimum Distance Estimators (MDE)*

MDE is developped by Wolfowitz[28]. In his paper he also proved MDE consistency. In this research, besides MLE , MDE is used to create several different estimators. The main idea in MDE is to fit the distribution to the sample data. MDE minimize a Goodness of fit statistics (GOF) between the distribution and the data values. The GOF quantifies the difference (each GOF differently) between the EDF and CDF. MDE has several advantages to the other estimation techniques.

- MD estimates are not very susceptible to outliers.[21:617]
- They are consistent.
- MD estimation methodology can be used to estimate the shape and scale parameters besides location.[21:616] Originally this technique was used only to estimate the location parameter.

Dr.A.H.MOORE and his students has studied extensively the MD methodology and its applications to different distributions. Their studied showed that Minimum distance estimation for Gamma and Weibull distribution gave better estimates than MLE. In their studies, they estimated all parameters by MLE then, sliding the distribution left and right to find the location parameter which minimizes the "Goodness-of-Fit " statistic. Once the location obtained by this way, the other parameter or parameters are reestimated by MLE.

### *3.6 Goodness-of-Fit Statistic*

A statistic measuring the difference between EDF and CDF is called GOF statistic based on EDF.. They measures the vertical differences between EDF and CDF. I will consider three different GOF statistic: Cramer–Von Mises (CvM), the Anderson–Darling statistic,called $A^2$ and Modified W statistic.

### *3.7 Cramer–von Mises family*

This class of measures discrepancy is given by Stephens as follows [25:100-101]

$$Q = n \int_{-\infty}^{\infty} [F_n(x) - F(x)]^2 \psi(x) dF(x) \tag{3.11}$$

When $\psi(x)$ is 1, the statistic is called Cramer–von Mises statistic and When $\psi(x) = [F(x)(1 - F(x))]^{-1}$ the statistic is Anderson-Darling statistic.[25:100-101] It is not very hard to see the computational difficulty of this formula. But, Stephens found suitable formulas for both CvM and $A^2$ by using the Probability Integral Transformation (PIT) as follows:

$$Cvm = \frac{1}{12n} + \frac{1}{n} \sum_{i=1}^{n} (Z_i - \frac{2i-1}{2n})^2 \tag{3.12}$$

$$A^2 = -n - 2 \sum_{i=1}^{n} \frac{2i-1}{2n} (\log_e(Z_i) + log_e(1 - Z_{n-i+1})) \tag{3.13}$$

In these equations ,Given a random sample $X_1 \ldots X_n$, $Z_i = F(X_i)$, i=1,...,n and $Z_i$ values arranged in ascending order,$Z_{(1)} < Z_{(2)} < \ldots < Z_n$. In this research, to prevent computational errors, $Z_i$ is bounded such that $0.0001 \leq Z_i \leq 0.9999$. The reasons to choose $A^2$ are numerous. Miller[19:26], Gallagher[8:40], Crown[4],and many others concluded that $A^2$ performed well in their studies. Also, Stephens among many GOF statistic including S [18],$\bar{Z}$ recommended $A^2$ for EDF tests with unknown parameters [25:167]

As an alternative test statistic, a modification of the W statistic [20:1375-1391] is chosen for testing the goodness of fit for the three parameter Weibull distribution.

### *3.8 Minimum Distance of Location*

The unknown parameters of the Weibull first estimated by MLE, then using these estimates, the location is varied to minimize the $A^2$ or Cvm statistic.

The golden search method is used to obtain minimum distance estimates of location when needed. An error tolerance of $10^{-6}$ is used.

### *3.9 Random Deviate Generation*

In this research, the Weibull data is transformed to the Extreme value distribution by using an appropriate transformation. Since the ML estimates are equalvariant with respect to location and scale and the extreme value distribution does have only location and scale parameters, only one set of parameters is used to obtain critical values and do a power study for each sample size. The true Weibull parameters are as follows:

- Location = 10,
- Scale = 4, and
- Shape = 3.

### *3.10 $\tilde{W}^*$ Test Statistic and Extreme Value Distribution*

Ozturk and Korukoglu in their paper suggested a new test statistic which is a modification of the W statistic and obtained as the ratio of two linear estimates of the scale parameter. They concluded that this new test statistic was computationally simple and had good power properties e.g. They shown that $\tilde{W}^*$ was more powerful than Shapiro and Brain's test,a test based on the $W_w statistic$[24].

*3.10.1 The Extreme Value Distribution.* One easily can notice that all these paper titles includes both the Weibull distribution and the Extreme Value Distribution. Because when the location is known, or 0, the following steps will show how to transform the Weibull data to the Extreme Value Distribution. [25:150]

- The Extreme value distribution CDF is as follows

$$F(y) = exp\left(-exp\left(\frac{-(y-\phi)}{\delta}\right)\right)$$

  $-\infty < y < \infty$ (3.14)

  with $\delta = 1/\beta$, where $\beta$ is the Weibull distribution shape parameter and $phi = -\ln(\theta)$, where $\theta$ is the Weibull distribution scale parameter.

  Using these parameter relationships, one can estimate the Weibull distributions parameters by MLE, then obtain the Extreme value Distribution parameters without estimating them by MLE.
- Make the transformation $Y_i = -\ln(X_i - \gamma), i = 1, ..., n$.
- Arrange the $Y_i$ in ascending order.
- Test that the Y−sample is from the extreme value distribution given the CDF above.

The extreme value distribution is one of the most used distributions modelling the extreme values of random events [20:1376] and has an extensive literature. Harter and Moore [9] reviewed the historical work for this distribution.

The extreme value distribution is used on modelling rainfall, flood flows, Rantz and Riggs, earthquakes, general meteorological data, air. .ft load, corrosion, and microorganism survival times. [14:274]

*3.10.2 $\tilde{W}^*$ test statistic.* This test statistic is based on the comparison of two different estimators of the scale parameters. In their study they used two different scale estimators which are linear unbiased estimators, one is $\hat{\sigma}$,the probability−weighted moment estiamtor of $\sigma$ and b, D'Agostino's estimator of $\sigma$ [5]. The formulas for b are as follows:

$$b = [0.6079 \sum_{i=1}^{n} w_{n+i} X_{(i)} - 0.2570 \sum_{i=1}^{n} w_i X_{(i)}]/n \tag{3.15}$$

where $w_i = ln[(n+1)/(n+1-i)]$ $i = 1, 2, ..., n-1$

$w_n = n - \sum_{i=1}^{n-1} w_i$

$w_{n+i} = w_i(1 + \ln(w_i)) - 1$ and $w_{2n} = 0.4228n - \sum_{i=1} n - 1w_{n=i}$.

$\hat{\sigma}$ can be written as $\sum_{i=1}^{n}(2j - n - 1)X_{(i)}/(0.693147n(n-1))$ This statistic is an unbiased estimator of $\sigma$ as shown by Ozturk. Then the proposed test statistic becomes

$$W^* = \frac{b}{\hat{\sigma}} \tag{3.16}$$

Later, they standardized the equation as follows;

$$\tilde{W}^* = \frac{W^* - 1.0 - \frac{0.13}{\sqrt{n}} + \frac{1.18}{n}}{\frac{0.49}{\sqrt{n}} - \frac{0.36}{n}} \tag{3.17}$$

### *3.11 Approach and Methodology*

The three parameter Weibull Distribution can be transformed easily to the two parameter Weibull Distribution by subtracting the location parameter out from all the ordered observations and shape parameter. After obtaining the two parameter Weibull distribution, I will transform it to the Extreme Value Distribution by taking the logarithm ( base e ) of the observations. Then using one of the modified test statistics I will test the hypothesis whether the sample comes from the Extreme or not. Indeed if I reject the null hypothesis, I will also reject that the original data comes from the two parameter Weibull Distribution.

In my research I will follow the steps below :

1) Find all three parameters of the Weibull Distribution by Maximum Likelihood Estimation ( MLE ).

2) Keep the Weibull's shape and scale parameter constant, then estimate the location parameter by Minimum Distance Estimation.

3) Re-estimate the Weibull's shape and scale parameter by MLE keeping the location parameter constant.

4) Take the observations, subtract location estimate from each of them.

5) Transform the data to the extreme value distribution.

6) Perform a goodness-of-fit test to check whether the transformed data come from the Extreme Value Distribution.

7) Generate tables of critical values of the new test statistic.

8) Perform a power study for the new test against many other classical distributions.

# *IV. Results*

## *4.1 Introduction*

In this chapter, the results of this thesis research will be presented including the critical value and power comparison tables.

## *4.2 Critical Values*

We will use the 5000 Weibull data samples' parameter estimates to obtain the AD and W test statistics. Later these 5000 values will be ranked and tables will be made of the 0.01, 0.05, 0.10, 0.15, and 0.20 testing significance ($\alpha$) level critical values for all sample sizes. The flowchart for calculating critical values is presented in table 4.1.

In this research the bootstrap method will be used to compute the critical values. In this method, 5000 test statistics plotted on the horizontal axis versus some plotting position on the vertical axis. For the plotting position in this thesis, the median rank approximation shown here will be used.

As expected the critical values for both statistic increased slightly as sample size increased. Besides the power study, an experiment designed to check invariance property of the Weibull MLEs. In this experiment, from each parameter 3 different values are chosen : for location (10,15,20), for shape (3,4,5), and for scale (3,4,5). Then the results showed that the critical values for a given sample were equal. From this result, I concluded that the 5000 repetitions was enough for this Monte Carlo simulation and the study could be done only using one set of parameters (in this research location=10,scale=4,and shape=3 are chosen ).

$$y_i = \frac{i - .3}{n + .4} \tag{4.1}$$

Anderson Darling Critical Values

| 1-x | n=10 | n=15 | n=20 | n=25 | n=30 | n=40 | n=50 |
|---|---|---|---|---|---|---|---|
| .20 | 0.372164 | 0.400007 | 0.406372 | 0.413902 | 0.422886 | 0.424601 | 0.426555 |
| .15 | 0.399923 | 0.431546 | 0.442253 | 0.448841 | 0.461475 | 0.461979 | 0.462581 |
| .10 | 0.439678 | 0.475123 | 0.488239 | 0.496711 | 0.511983 | 0.519688 | 0.520989 |
| .05 | 0.500419 | 0.560200 | 0.584271 | 0.592173 | 0.616696 | 0.617272 | 0.617850 |
| .01 | 0.632765 | 0.712128 | 0.790437 | 0.817037 | 0.824887 | 0.831194 | 0.836791 |

Table 4.1. In this table, Only MLE is used and AD statistic is minimized

Anderson Darling Critical Values

| 1-x | n=10 | n=15 | n=20 | n=25 | n=30 | n=40 | n=50 |
|---|---|---|---|---|---|---|---|
| .20 | 0.369178 | 0.397664 | 0.403394 | 0.411826 | 0.422235 | 0.426182 | 0.427678 |
| .15 | 0.397186 | 0.428653 | 0.439700 | 0.447060 | 0.459889 | 0.465496 | 0.466856 |
| .10 | 0.437097 | 0.473313 | 0.486487 | 0.494679 | 0.510844 | 0.518742 | 0.521566 |
| .05 | 0.499525 | 0.556969 | 0.580953 | 0.590656 | 0.612755 | 0.615322 | 0.618043 |
| .01 | 0.630202 | 0.713775 | 0.785557 | 0.813947 | 0.819577 | 0.828025 | 0.836470 |

Table 4.2. In this table, MLE and MD is used and AD statistic is minimized

The plotting position values on the vertical axis presents a scale between zero and one which represents percentiles. The 80th, 85th, 90th, and 99th percentiles are obtained by interpolating between the two plotted points whose vertical axis values surround the respective percentile value. Table 4.1 can be used when the minimum distance is not calculated. Table 4.2 is prepared as shown in Chapter 3 Methodology section using the AD statistic. Table 4.3 can be used when the minimum distance is not calculated.

Table 4.4 is prepared as shown in Chapter 3 Methodology section using the W statistic.

### 4.3 Power Study

In this section the results of the power study will be presented.

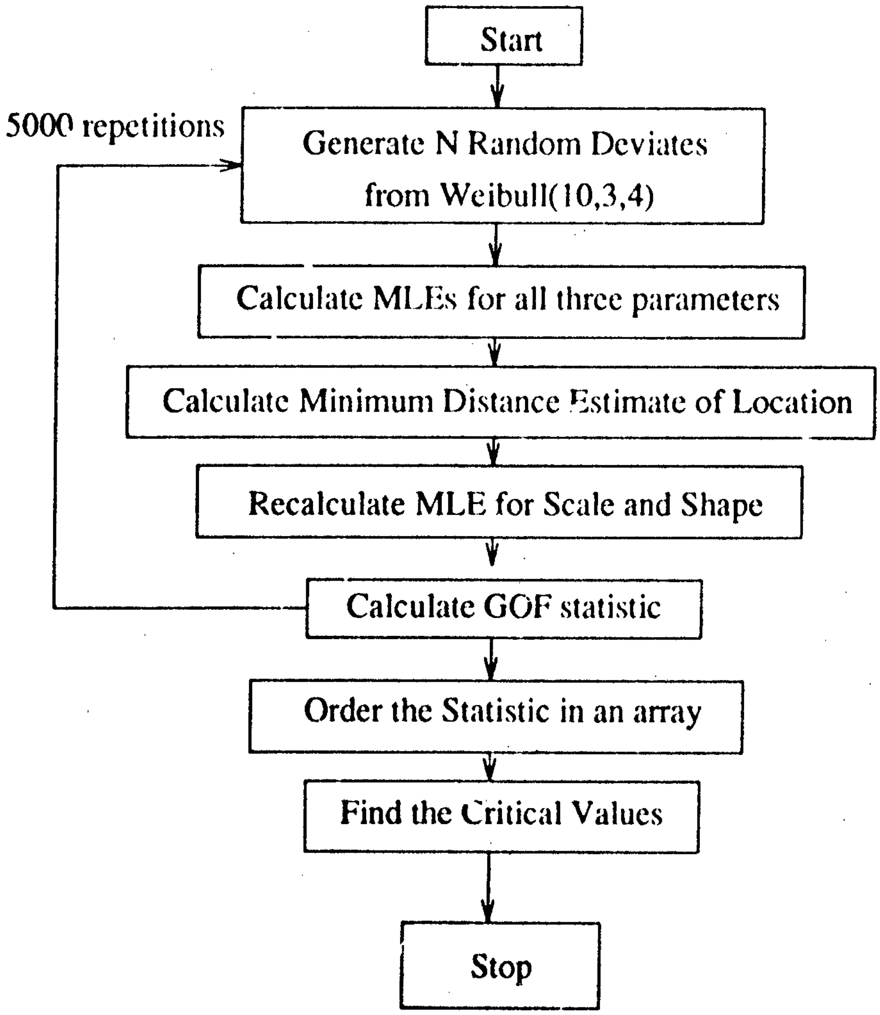


Figure 4.1. Generation of the critical values

W statistic Critical Values (only MLE is used )

| 1-x | n=10 | n=15 | n=20 | n=25 | n=30 | n=40 | n=50 |
|---|---|---|---|---|---|---|---|
| .900 | 0.065181 | 0.145342 | 0.185332 | 0.227954 | 0.281339 | 0.310155 | 0.315440 |
| .925 | 0.127089 | 0.214784 | 0.273520 | 0.302680 | 0.362434 | 0.399650 | 0.410232 |
| .950 | 0.201946 | 0.315025 | 0.383236 | 0.398015 | 0.492216 | 0.513646 | 0.539491 |
| .975 | 0.335798 | 0.485980 | 0.557636 | 0.612199 | 0.703691 | 0.720623 | 0.718460 |
| .995 | 0.612513 | 0.840787 | 0.942273 | 1.078666 | 1.040576 | 1.125672 | 1.170674 |
| .100 | -0.853239 | -0.944861 | -0.969439 | -0.969083 | -0.943755 | -0.932115 | -0.912232 |
| .075 | -0.908244 | -1.012667 | -1.037624 | -1.052842 | -1.022346 | -1.028218 | -0.992610 |
| .050 | -0.979875 | -1.090568 | -1.122980 | -1.162780 | -1.134543 | -1.120286 | -1.096457 |
| .025 | -1.118930 | -1.257186 | -1.274248 | -1.308831 | -1.289544 | -1.306388 | -1.280124 |
| .005 | -1.309019 | -1.525660 | -1.546312 | -1.643209 | -1.612966 | -1.612228 | -1.579373 |

Table 4.3. In this table, only MLE and W statistic are used

W statistic Critical Values (MLE and MDE are used)

| 1-x | n=10 | n=15 | n=20 | n=25 | n=30 | n=40 | n=50 |
|---|---|---|---|---|---|---|---|
| .900 | 0.071596 | 0.150106 | 0.186432 | 0.227004 | 0.284234 | 0.310347 | 0.317900 |
| .925 | 0.132874 | 0.218182 | 0.275954 | 0.304353 | 0.366422 | 0.400847 | 0.414499 |
| .950 | 0.209956 | 0.319090 | 0.384488 | 0.409164 | 0.497321 | 0.518350 | 0.540453 |
| .975 | 0.337767 | 0.488484 | 0.564401 | 0.622717 | 0.707745 | 0.722411 | 0.724813 |
| .995 | 0.616232 | 0.844150 | 0.948521 | 1.085643 | 1.052511 | 1.138178 | 1.196816 |
| .100 | -0.809158 | -0.908838 | -0.930482 | -0.937860 | -0.927955 | -0.922159 | -0.909192 |
| .075 | -0.868837 | -0.966370 | -0.991926 | -1.008289 | -0.997781 | -1.012072 | -0.982636 |
| .050 | -0.934449 | -1.039731 | -1.070275 | -1.104532 | -1.091558 | -1.098467 | -1.086252 |
| .025 | -1.041221 | -1.182828 | -1.206187 | -1.241062 | -1.234929 | -1.275192 | -1.263546 |
| .005 | -1.245098 | -1.414740 | -1.461238 | -1.538003 | -1.552484 | -1.558163 | -1.529731 |

Table 4.4. In this table, MLE and MDE and W statistic are used

After obtaining the tables of critical values, by generating random numbers from selected distributions a power study can be made. We can test a random sample given the Null hypothesis that the random sample is from a Weibull Distribution with estimated parameters versus the alternative hypothesis that the data is from the distribution used to generate the random sample tested. The A-D test statistic can be obtained first estimating all three parameters by MLE, then obtaing the location estimate by minimizing this statistic given the scale and shape estimates, finally keeping the location estimate constant, reestimating the shape and scale estimates. Later, this statistic should be compared to the appropriate critical values. If it is larger than the critical value compared, then it should be concluded that the sample is not from the Weibull distribution, and the Null hypothesis is rejected. This test will be done for sample sizes of 10,15,20,25,30,40,50 with 5000 of each case. When we divide the number of rejections by 5000 (total number of samples), we will obtain the power of the test. This power can be compared by only the well known Chi-Square test. Because in the litterature there is no test for the three parameter Weibull with all parameters unknown. In order to make a comparison of the test based on the A-D statistic, only one competitor was chosen and that was Ozturk's standardized $\tilde{W}^*$ statistic. The tables show the hypothesized Weibull distribution with shape equal to 3, scale equal 4 and location to 10, and the alternative distributions with level of significance of 0.05 and 0.01. The distributions are as follows:

1. Weibull with shape = 3.0, scale = 4.0 and location = 10.0
2. Uniform on interval (10,15)
3. Uniform on interval (8,12)
4. Gamma with shape = 1.0 , scale =.2 and location = 10.0
5. Gamma with shape = 2.0 , scale =.2 and location = 10.0
6. Gamma with shape = 3.0 , scale =.2 and location = 10.0
7. Normal with mean = 15.0 and variance = 2.0

8. Normal with mean =12.0 and variance = 1.0

9. Beta with p = 2 and q = 2

First, the tables computed by using the standardized $\tilde{W}*$ statistic will be presented. Then tables computed by minimizing the A-D statistic will follow these tables. Since there is no prior power information about the three parameter unknown case, I also obtained the the tables and critical values by only using the MLE.

In this study, When shape parameter was 1 or less than 1 the sample rejected. Because when location equals to $x_{(1)}$ Likelihood function become infinite. We have to ignore this because all data are actually discrete, and the singularity disappears on taking this account [26:360]

### *4.4 Verification and Validation*

The computer code is verified line by line extensively. All the random number generators are found in Banks and Carson [1:294-300]. The random generator is chosen very carefully. In order not to have any randomization problems such as, numbers do not appear to be distributed uniformly on [0,1] or they explicit correlation with each other, a recommended IMSL generator is used. To validate the computer code 1000 samples is taken and stored in two different files, then using Mathcad the results are confirmed using several different approaches such as the partials should be close to zero, the Mathcad estimates and estimates obtained by using Harter and Moore's algorithm should be close to each other.

The A-D statistic, $\tilde{W}*$ statistic, and Extreme Value Transformation procedures are checked in the same way using Mathcad. By doing this It is intended to check the validity of the computer code and to present a way for the future practitioners to learn the theoretical concepts easily .

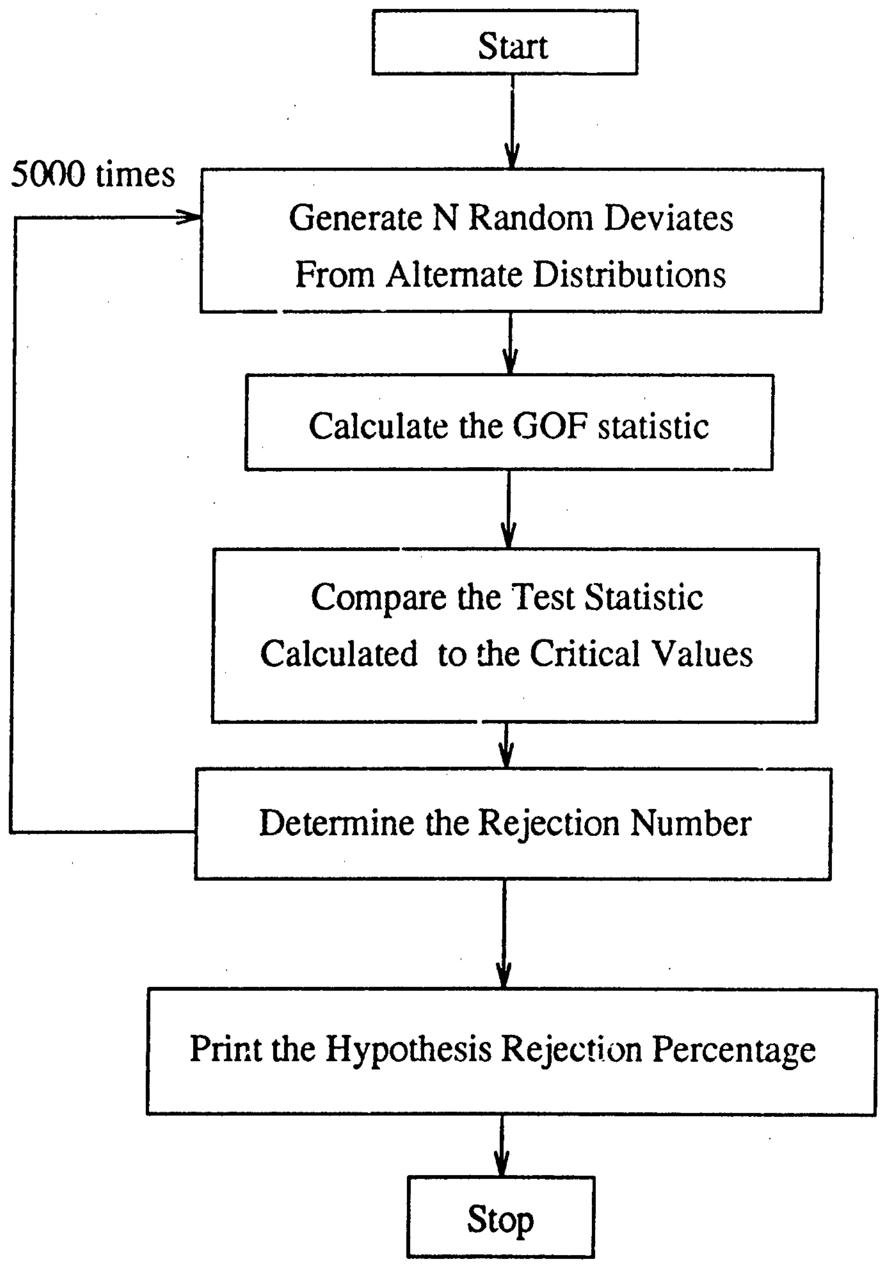


Figure 4.2. Power Study

$H_0$:Weibull Distribution ; $H_a$:Another Distribution

| Sample Size | 1-x | Weibull Shape=3 | Uniform U(8,12) | Uniform U(10-15) | Gamma Shape=3.0 | Gamma Shape=4.0 |
|---|---|---|---|---|---|---|
| 10 | .20 | 0.191400 | 0.285400 | 0.286000 | 0.216200 | 0.204600 |
| | .15 | 0.143400 | 0.220600 | 0.224000 | 0.156000 | 0.154400 |
| | .10 | 0.097200 | 0.148600 | 0.146200 | 0.107400 | 0.108600 |
| | .05 | 0.049400 | 0.081000 | 0.078800 | 0.059000 | 0.057000 |
| | .01 | 0.015000 | 0.016200 | 0.016200 | 0.012200 | 0.011600 |
| 15 | .20 | 0.182800 | 0.379000 | 0.373600 | 0.191400 | 0.208000 |
| | .15 | 0.137400 | 0.312400 | 0.302800 | 0.138400 | 0.155400 |
| | .10 | 0.093200 | 0.233800 | 0.224200 | 0.091200 | 0.104800 |
| | .05 | 0.045000 | 0.130200 | 0.120600 | 0.038800 | 0.047400 |
| | .01 | 0.012800 | 0.039800 | 0.034400 | 0.008800 | 0.010800 |
| 20 | .20 | 0.206000 | 0.508600 | 0.502000 | 0.207200 | 0.224000 |
| | .15 | 0.150400 | 0.434800 | 0.421000 | 0.156200 | 0.170800 |
| | .10 | 0.108000 | 0.346800 | 0.326000 | 0.110600 | 0.116400 |
| | .05 | 0.046800 | 0.200800 | 0.191800 | 0.049600 | 0.052000 |
| | .01 | 0.007800 | 0.054200 | 0.049000 | 0.009600 | 0.010400 |
| 25 | .20 | 0.194400 | 0.601400 | 0.593000 | 0.212000 | 0.229200 |
| | .15 | 0.151000 | 0.529600 | 0.513400 | 0.160000 | 0.181400 |
| | .10 | 0.109200 | 0.436400 | 0.424600 | 0.106800 | 0.128800 |
| | .05 | 0.052400 | 0.285400 | 0.271200 | 0.048400 | 0.063000 |
| | .01 | 0.009600 | 0.086600 | 0.084600 | 0.007600 | 0.013400 |
| 30 | .20 | 0.184400 | 0.671600 | 0.662200 | 0.204000 | 0.222600 |
| | .15 | 0.139200 | 0.602800 | 0.585600 | 0.147200 | 0.169400 |
| | .10 | 0.098800 | 0.514400 | 0.493200 | 0.098400 | 0.118800 |
| | .05 | 0.050200 | 0.350400 | 0.320400 | 0.039800 | 0.053600 |
| | .01 | 0.013000 | 0.138800 | 0.125400 | 0.006600 | 0.010200 |
| 40 | .20 | 0.189600 | 0.802800 | 0.798200 | 0.222800 | 0.231400 |
| | .15 | 0.146800 | 0.749600 | 0.739000 | 0.165000 | 0.178800 |
| | .10 | 0.099200 | 0.668800 | 0.654800 | 0.108400 | 0.127600 |
| | .05 | 0.046800 | 0.524200 | 0.514400 | 0.051600 | 0.066200 |
| | .01 | 0.009400 | 0.235800 | 0.222200 | 0.010000 | 0.013200 |
| 50 | .20 | 0.194000 | 0.896800 | 0.893600 | 0.231400 | 0.261000 |
| | .15 | 0.149200 | 0.859600 | 0.857600 | 0.180400 | 0.206800 |
| | .10 | 0.092800 | 0.787000 | 0.784600 | 0.114000 | 0.145000 |
| | .05 | 0.045000 | 0.662400 | 0.662800 | 0.053200 | 0.081600 |
| | .01 | 0.011200 | 0.392600 | 0.380600 | 0.011200 | 0.024000 |

Table 4.5. In this table, Only MLE is used to estimate all three parameters and AD statistic is used as a GOF statistic

$H_0$:Weibull Distribution; $H_a$:Another Distribution

| Sample Size | 1-x | Gamma Shape=5 | Normal N(15,2) | Normal N(12,1) | Beta B(2,2) |
|---|---|---|---|---|---|
| 10 | .20 | 0.226200 | 0.201400 | 0.185800 | 0.193400 |
| | .15 | 0.174400 | 0.149800 | 0.140200 | 0.148400 |
| | .10 | 0.121600 | 0.096600 | 0.087800 | 0.097800 |
| | .05 | 0.061600 | 0.045400 | 0.044400 | 0.048600 |
| | .01 | 0.014800 | 0.010800 | 0.010200 | 0.013200 |
| 15 | .20 | 0.220800 | 0.194000 | 0.183400 | 0.204200 |
| | .15 | 0.170200 | 0.148400 | 0.144600 | 0.153400 |
| | .10 | 0.120800 | 0.099200 | 0.091400 | 0.102200 |
| | .05 | 0.057000 | 0.047800 | 0.041800 | 0.043000 |
| | .01 | 0.014800 | 0.013000 | 0.009400 | 0.008400 |
| 20 | .20 | 0.245200 | 0.208600 | 0.201000 | 0.275200 |
| | .15 | 0.194200 | 0.153200 | 0.147600 | 0.210000 |
| | .10 | 0.135200 | 0.109400 | 0.105400 | 0.145400 |
| | .05 | 0.064000 | 0.049000 | 0.045800 | 0.059600 |
| | .01 | 0.011200 | 0.008000 | 0.008600 | 0.007400 |
| 25 | .20 | 0.251800 | 0.195800 | 0.198000 | 0.280800 |
| | .15 | 0.198600 | 0.147400 | 0.150000 | 0.224600 |
| | .10 | 0.148400 | 0.102800 | 0.101400 | 0.159000 |
| | .05 | 0.070800 | 0.048200 | 0.050000 | 0.077000 |
| | .01 | 0.014600 | 0.009600 | 0.009200 | 0.012600 |
| 30 | .20 | 0.241600 | 0.198000 | 0.192000 | 0.309400 |
| | .15 | 0.186800 | 0.151400 | 0.147000 | 0.237000 |
| | .10 | 0.130200 | 0.103800 | 0.097600 | 0.169400 |
| | .05 | 0.060800 | 0.050400 | 0.043000 | 0.082800 |
| | .01 | 0.015000 | 0.010600 | 0.007800 | 0.017600 |
| 40 | .20 | 0.258600 | 0.206200 | 0.199000 | 0.378000 |
| | .15 | 0.198600 | 0.158000 | 0.152000 | 0.309000 |
| | .10 | 0.145600 | 0.100400 | 0.094200 | 0.223800 |
| | .05 | 0.081600 | 0.047800 | 0.044800 | 0.126800 |
| | .01 | 0.015200 | 0.006600 | 0.008200 | 0.024800 |
| 50 | .20 | 0.274600 | 0.211000 | 0.211800 | 0.442800 |
| | .15 | 0.223200 | 0.166400 | 0.165800 | 0.363200 |
| | .10 | 0.158000 | 0.104200 | 0.107600 | 0.270200 |
| | .05 | 0.088800 | 0.055200 | 0.053000 | 0.154800 |
| | .01 | 0.025000 | 0.012200 | 0.010400 | 0.043000 |

Table 4.6. In this table, Only MLE are used and AD statistic is minimized

| $H_0$:Weibull Distribution ; $H_a$:Another Distribution |
|---|

| Sample Size | 1-x | Weibull Shape=3 | Uniform U(8,12) | Uniform U(10-15) | Gamma Shape=3.0 | Gamma Shape=4.0 |
|---|---|---|---|---|---|---|
| 10 | .20 | 0.191000 | 0.288800 | 0.289000 | 0.205600 | 0.195600 |
| | .15 | 0.142400 | 0.220800 | 0.224800 | 0.157200 | 0.147400 |
| | .10 | 0.096600 | 0.148000 | 0.148600 | 0.103200 | 0.097000 |
| | .05 | 0.049600 | 0.079800 | 0.078200 | 0.048800 | 0.049600 |
| | .01 | 0.015200 | 0.016200 | 0.016800 | 0.009200 | 0.010400 |
| 15 | .20 | 0.183400 | 0.375000 | 0.373600 | 0.179400 | 0.198600 |
| | .15 | 0.138200 | 0.312000 | 0.303600 | 0.136000 | 0.152800 |
| | .10 | 0.093400 | 0.231400 | 0.221600 | 0.090800 | 0.105400 |
| | .05 | 0.044600 | 0.131200 | 0.122600 | 0.040200 | 0.048800 |
| | .01 | 0.012400 | 0.038200 | 0.033000 | 0.007400 | 0.011200 |
| 20 | .20 | 0.208000 | 0.510000 | 0.504000 | 0.188400 | 0.230200 |
| | .15 | 0.150200 | 0.433000 | 0.423400 | 0.137600 | 0.174400 |
| | .10 | 0.106200 | 0.344000 | 0.326600 | 0.087800 | 0.122200 |
| | .05 | 0.047400 | 0.199800 | 0.192000 | 0.040200 | 0.062200 |
| | .01 | 0.008000 | 0.053800 | 0.049800 | 0.007000 | 0.019000 |
| 25 | .20 | 0.196600 | 0.601000 | 0.592400 | 0.196000 | 0.223000 |
| | .15 | 0.150600 | 0.527600 | 0.515200 | 0.142600 | 0.168400 |
| | .10 | 0.108800 | 0.435800 | 0.425000 | 0.097000 | 0.121000 |
| | .05 | 0.052200 | 0.282200 | 0.268600 | 0.042400 | 0.062600 |
| | .01 | 0.009600 | 0.084600 | 0.083200 | 0.005000 | 0.016200 |
| 30 | .20 | 0.183800 | 0.668200 | 0.672000 | 0.196400 | 0.208400 |
| | .15 | 0.139800 | 0.602200 | 0.592200 | 0.145400 | 0.154800 |
| | .10 | 0.098200 | 0.510200 | 0.500000 | 0.092400 | 0.106600 |
| | .05 | 0.050400 | 0.352400 | 0.336000 | 0.039200 | 0.052000 |
| | .01 | 0.013200 | 0.137600 | 0.128800 | 0.008600 | 0.012000 |
| 40 | .20 | 0.189000 | 0.800400 | 0.801600 | 0.212800 | 0.226800 |
| | .15 | 0.145200 | 0.747000 | 0.743200 | 0.156000 | 0.174800 |
| | .10 | 0.099400 | 0.665400 | 0.659200 | 0.100600 | 0.122200 |
| | .05 | 0.047000 | 0.523000 | 0.508800 | 0.049200 | 0.068800 |
| | .01 | 0.009400 | 0.234200 | 0.216800 | 0.009600 | 0.016800 |
| 50 | .20 | 0.193400 | 0.895000 | 0.891800 | 0.215000 | 0.248000 |
| | .15 | 0.149200 | 0.859400 | 0.855800 | 0.153200 | 0.202200 |
| | .10 | 0.093200 | 0.785600 | 0.782400 | 0.098400 | 0.142000 |
| | .05 | 0.044400 | 0.658200 | 0.661400 | 0.049700 | 0.077000 |
| | .01 | 0.011000 | 0.389000 | 0.377800 | 0.009800 | 0.023200 |

Table 4.7. In this table, MLE and MD is used and AD statistic is minimized

| $H_0$:Weibull Distribution; $H_a$:Another Distribution | | | | | |
|---|---|---|---|---|---|
| Sample Size | 1-x | Gamma Shape=5 | Normal N(15,2) | Normal N(12,1) | Beta B(2,2) |
| 10 | .20 | 0.200800 | 0.199000 | 0.202200 | 0.195600 |
| | .15 | 0.154800 | 0.145000 | 0.148600 | 0.145400 |
| | .10 | 0.103800 | 0.089400 | 0.100200 | 0.096000 |
| | .05 | 0.055600 | 0.044600 | 0.052200 | 0.049800 |
| | .01 | 0.009400 | 0.010000 | 0.013600 | 0.012200 |
| 15 | .20 | 0.201200 | 0.195600 | 0.183000 | 0.206400 |
| | .15 | 0.155400 | 0.156000 | 0.135200 | 0.156400 |
| | .10 | 0.105600 | 0.104000 | 0.090200 | 0.103600 |
| | .05 | 0.047400 | 0.047800 | 0.039600 | 0.047600 |
| | .01 | 0.012800 | 0.008200 | 0.008800 | 0.010000 |
| 20 | .20 | 0.246200 | 0.217400 | 0.211200 | 0.300000 |
| | .15 | 0.194000 | 0.170600 | 0.160800 | 0.233400 |
| | .10 | 0.131800 | 0.122200 | 0.114200 | 0.167200 |
| | .05 | 0.065600 | 0.056800 | 0.054600 | 0.079400 |
| | .01 | 0.013400 | 0.012800 | 0.010200 | 0.014600 |
| 25 | .20 | 0.233800 | 0.206600 | 0.195200 | 0.308100 |
| | .15 | 0.183800 | 0.157400 | 0.148600 | 0.239200 |
| | .10 | 0.133000 | 0.108200 | 0.098800 | 0.169500 |
| | .05 | 0.062200 | 0.050600 | 0.047400 | 0.081000 |
| | .01 | 0.011800 | 0.009600 | 0.007000 | 0.014800 |
| 30 | .20 | 0.242000 | 0.203600 | 0.190000 | 0.313800 |
| | .15 | 0.191600 | 0.157000 | 0.142800 | 0.244800 |
| | .10 | 0.129600 | 0.105600 | 0.095600 | 0.176000 |
| | .05 | 0.064000 | 0.052000 | 0.042800 | 0.084800 |
| | .01 | 0.013200 | 0.011600 | 0.009400 | 0.015800 |
| 40 | .20 | 0.253000 | 0.208400 | 0.211400 | 0.376200 |
| | .15 | 0.192400 | 0.158600 | 0.156800 | 0.299200 |
| | .10 | 0.138800 | 0.109400 | 0.105600 | 0.219800 |
| | .05 | 0.076200 | 0.054600 | 0.050000 | 0.126200 |
| | .01 | 0.013800 | 0.013200 | 0.008200 | 0.025400 |
| 50 | .20 | 0.261400 | 0.212000 | 0.208000 | 0.439800 |
| | .15 | 0.209400 | 0.162400 | 0.159800 | 0.369400 |
| | .10 | 0.147600 | 0.110200 | 0.107600 | 0.275600 |
| | .05 | 0.083800 | 0.054400 | 0.056000 | 0.160200 |
| | .01 | 0.021400 | 0.011000 | 0.011200 | 0.044200 |

Table 4.8. In this table, MLE and MD are used and AD statistic is minimized

$H_0$:Weibull Distribution ; $H_a$:Another Distribution

| Sample Size | 1-x | Weibull Shape=3 | Uniform U(8,12) | Uniform U(10-15) | Gamma Shape=3.0 | Gamma Shape=4.0 |
|---|---|---|---|---|---|---|
| 10 | .20 | 0.203600 | 0.233400 | 0.221000 | 0.216200 | 0.204600 |
| | .15 | 0.156000 | 0.185400 | 0.175000 | 0.156000 | 0.154400 |
| | .10 | 0.109600 | 0.139200 | 0.126000 | 0.107400 | 0.108600 |
| | .05 | 0.051200 | 0.074200 | 0.065400 | 0.059000 | 0.057000 |
| | .01 | 0.011600 | 0.026600 | 0.018800 | 0.012200 | 0.011600 |
| 15 | .20 | 0.191400 | 0.319800 | 0.292800 | 0.231000 | 0.248200 |
| | .15 | 0.145000 | 0.265200 | 0.239200 | 0.175600 | 0.196600 |
| | .10 | 0.098600 | 0.209400 | 0.186200 | 0.119600 | 0.116200 |
| | .05 | 0.045600 | 0.124200 | 0.095200 | 0.049400 | 0.074400 |
| | .01 | 0.010400 | 0.038800 | 0.027800 | 0.009000 | 0.017000 |
| 20 | .20 | 0.185000 | 0.423600 | 0.391600 | 0.274000 | 0.285000 |
| | .15 | 0.140800 | 0.372600 | 0.339200 | 0.210000 | 0.222600 |
| | .10 | 0.101000 | 0.308800 | 0.275200 | 0.146000 | 0.167200 |
| | .05 | 0.053800 | 0.209600 | 0.170200 | 0.074400 | 0.090[illegible]00 |
| | .01 | 0.009600 | 0.083600 | 0.068200 | 0.015200 | 0.023200 |
| 25 | .20 | 0.190000 | 0.531600 | 0.508600 | 0.310200 | 0.290800 |
| | .15 | 0.144800 | 0.461800 | 0.431400 | 0.234600 | 0.236200 |
| | .10 | 0.094000 | 0.380200 | 0.345000 | 0.160200 | 0.170000 |
| | .05 | 0.045600 | 0.282000 | 0.243400 | 0.076600 | 0.096600 |
| | .01 | 0.007800 | 0.115200 | 0.083600 | 0.012400 | 0.020400 |
| 30 | .20 | 0.195800 | 0.659000 | 0.635200 | 0.316600 | 0.303200 |
| | .15 | 0.151800 | 0.592800 | 0.560600 | 0.247200 | 0.244600 |
| | .10 | 0.097800 | 0.501200 | 0.467400 | 0.166000 | 0.174600 |
| | .05 | 0.048200 | 0.377800 | 0.343000 | 0.059200 | 0.091800 |
| | .01 | 0.012400 | 0.181800 | 0.139800 | 0.007000 | 0.019800 |
| 40 | .20 | 0.185200 | 0.820800 | 0.808800 | 0.351800 | 0.337400 |
| | .15 | 0.137200 | 0.755600 | 0.740600 | 0.279400 | 0.273600 |
| | .10 | 0.098600 | 0.683600 | 0.665600 | 0.215600 | 0.214000 |
| | .05 | 0.047000 | 0.531600 | 0.510400 | 0.110800 | 0.122200 |
| | .01 | 0.009400 | 0.310600 | 0.283600 | 0.027000 | 0.038800 |
| 50 | .20 | 0.193000 | 0.923600 | 0.917600 | 0.354400 | 0.372200 |
| | .15 | 0.143000 | 0.884200 | 0.875600 | 0.295000 | 0.307600 |
| | .10 | 0.093400 | 0.827600 | 0.818200 | 0.221800 | 0.241400 |
| | .05 | 0.049400 | 0.704800 | 0.694200 | 0.135000 | 0.154600 |
| | .01 | 0.010200 | 0.469800 | 0.444400 | 0.038000 | 0.051200 |

Table 4.9. In this table, Only MLE is and W statistic is used

| $H_0$:Weibull Distribution; $H_a$:Another Distribution | | | | | |
|---|---|---|---|---|---|
| Sample Size | 1-x | Gamma Shape=5 | Normal N(15,2) | Normal N(12,1) | Beta B(2,2) |
| 10 | .20 | 0.235400 | 0.226400 | 0.195200 | 0.158000 |
| | .15 | 0.187200 | 0.173800 | 0.148000 | 0.111000 |
| | .10 | 0.138400 | 0.124200 | 0.096800 | 0.068800 |
| | .05 | 0.075600 | 0.060200 | 0.048200 | 0.024000 |
| | .01 | 0.020000 | 0.015000 | 0.009600 | 0.003800 |
| 15 | .20 | 0.262600 | 0.214000 | 0.206400 | 0.144200 |
| | .15 | 0.213400 | 0.166200 | 0.155000 | 0.097600 |
| | .10 | 0.153000 | 0.114600 | 0.102600 | 0.061800 |
| | .05 | 0.076400 | 0.054800 | 0.046400 | 0.019200 |
| | .01 | 0.019000 | 0.014000 | 0.009600 | 0.001000 |
| 20 | .20 | 0.280000 | 0.217600 | 0.210200 | 0.193400 |
| | .15 | 0.220400 | 0.169600 | 0.151400 | 0.140600 |
| | .10 | 0.162600 | 0.114800 | 0.102800 | 0.096600 |
| | .05 | 0.094800 | 0.060800 | 0.051200 | 0.049000 |
| | .01 | 0.027200 | 0.018200 | 0.012800 | 0.007600 |
| 25 | .20 | 0.298400 | 0.217600 | 0.200600 | 0.208600 |
| | .15 | 0.238400 | 0.169600 | 0.148800 | 0.155800 |
| | .10 | 0.174200 | 0.118400 | 0.101600 | 0.093200 |
| | .05 | 0.097200 | 0.059200 | 0.048600 | 0.041200 |
| | .01 | 0.027000 | 0.014200 | 0.008000 | 0.001800 |
| 30 | .20 | 0.298400 | 0.210400 | 0.195800 | 0.272200 |
| | .15 | 0.243600 | 0.162400 | 0.149600 | 0.201600 |
| | .10 | 0.180600 | 0.112200 | 0.098600 | 0.132800 |
| | .05 | 0.094400 | 0.058600 | 0.049400 | 0.063000 |
| | .01 | 0.020600 | 0.017000 | 0.010600 | 0.006600 |
| 40 | .20 | 0.318000 | 0.222800 | 0.202800 | 0.357600 |
| | .15 | 0.256000 | 0.168600 | 0.155600 | 0.269000 |
| | .10 | 0.195200 | 0.117400 | 0.109600 | 0.198400 |
| | .05 | 0.120800 | 0.059600 | 0.056600 | 0.092000 |
| | .01 | 0.039000 | 0.015000 | 0.012800 | 0.016200 |
| 50 | .20 | 0.347200 | 0.254600 | 0.233800 | 0.461000 |
| | .15 | 0.290200 | 0.195600 | 0.177400 | 0.374400 |
| | .10 | 0.224800 | 0.134600 | 0.119200 | 0.277400 |
| | .05 | 0.144800 | 0.078800 | 0.066800 | 0.134400 |
| | .01 | 0.050600 | 0.016600 | 0.013200 | 0.030000 |

Table 4.10. In this table, Only MLE is and W statistic is used

| $H_0$:Weibull Distribution ; $H_a$:Another Distribution | | | | | | |
|---|---|---|---|---|---|---|
| Sample Size | 1-x | Weibull Shape=3 | Uniform U(8,12) | Uniform U(10-15) | Gamma Shape=3.0 | Gamma Shape=4.0 |
| 10 | .20 | 0.203600 | 0.233400 | 0.221000 | 0.219200 | 0.240800 |
| | .15 | 0.156000 | 0.185400 | 0.175000 | 0.161800 | 0.184200 |
| | .10 | 0.109600 | 0.139200 | 0.126000 | 0.113000 | 0.133800 |
| | .05 | 0.051200 | 0.074200 | 0.065400 | 0.055800 | 0.071200 |
| | .01 | 0.011600 | 0.026600 | 0.018800 | 0.008600 | 0.015600 |
| 15 | .20 | 0.191400 | 0.319800 | 0.298800 | 0.200600 | 0.235800 |
| | .15 | 0.145000 | 0.265200 | 0.249400 | 0.151400 | 0.185600 |
| | .10 | 0.098600 | 0.209400 | 0.190400 | 0.104800 | 0.132000 |
| | .05 | 0.045600 | 0.124200 | 0.107000 | 0.049400 | 0.069600 |
| | .01 | 0.010400 | 0.038800 | 0.030800 | 0.008200 | 0.017800 |
| 20 | .20 | 0.188000 | 0.445800 | 0.411800 | 0.223200 | 0.299400 |
| | .15 | 0.142000 | 0.390000 | 0.355200 | 0.163200 | 0.244800 |
| | .10 | 0.101000 | 0.327000 | 0.294000 | 0.111800 | 0.186000 |
| | .05 | 0.054200 | 0.224800 | 0.185400 | 0.055000 | 0.109400 |
| | .01 | 0.009000 | 0.089000 | 0.073200 | 0.009800 | 0.033600 |
| 25 | .20 | 0.193400 | 0.551400 | 0.527600 | 0.245200 | 0.285800 |
| | .15 | 0.145400 | 0.485400 | 0.460200 | 0.189200 | 0.229600 |
| | .10 | 0.096000 | 0.406600 | 0.373800 | 0.125800 | 0.167800 |
| | .05 | 0.045400 | 0.302000 | 0.268000 | 0.059000 | 0.092800 |
| | .01 | 0.009400 | 0.129600 | 0.102000 | 0.007200 | 0.022200 |
| 30 | .20 | 0.195400 | 0.668200 | 0.652400 | 0.288600 | 0.289400 |
| | .15 | 0.150400 | 0.605600 | 0.588000 | 0.222800 | 0.229000 |
| | .10 | 0.097000 | 0.525400 | 0.503000 | 0.147400 | 0.159200 |
| | .05 | 0.050000 | 0.406200 | 0.373600 | 0.071400 | 0.093200 |
| | .01 | 0.012400 | 0.187200 | 0.151200 | 0.012000 | 0.028400 |
| 40 | .20 | 0.186600 | 0.826000 | 0.815000 | 0.320600 | 0.316200 |
| | .15 | 0.136600 | 0.765200 | 0.742800 | 0.242200 | 0.254200 |
| | .10 | 0.096400 | 0.695800 | 0.676000 | 0.172600 | 0.195600 |
| | .05 | 0.046400 | 0.551200 | 0.529000 | 0.080600 | 0.115200 |
| | .01 | 0.008200 | 0.330800 | 0.294000 | 0.019000 | 0.035800 |
| 50 | .20 | 0.191600 | 0.924800 | 0.919800 | 0.330400 | 0.361200 |
| | .15 | 0.144200 | 0.888200 | 0.881800 | 0.256800 | 0.293200 |
| | .10 | 0.094000 | 0.832600 | 0.825000 | 0.189200 | 0.222600 |
| | .05 | 0.048200 | 0.716200 | 0.705000 | 0.081000 | 0.138200 |
| | .01 | 0.009400 | 0.499400 | 0.470800 | 0.019800 | 0.048200 |

Table 4.11. In this table, MLE and MDE and W statistic are used

$H_0$:Weibull Distribution; $H_a$:Another Distribution

| Sample Size | 1-x | Gamma Shape=5 | Normal N(15,2) | Normal N(12,1) | Beta B(2,2) |
|---|---|---|---|---|---|
| 10 | .20 | 0.231600 | 0.213400 | 0.204600 | 0.176400 |
| | .15 | 0.176800 | 0.163000 | 0.156400 | 0.126600 |
| | .10 | 0.130200 | 0.110400 | 0.103200 | 0.082200 |
| | .05 | 0.070000 | 0.062000 | 0.054600 | 0.039600 |
| | .01 | 0.012600 | 0.012000 | 0.010600 | 0.004600 |
| 15 | .20 | 0.240600 | 0.210000 | 0.191800 | 0.154000 |
| | .15 | 0.191000 | 0.167400 | 0.142400 | 0.113000 |
| | .10 | 0.138600 | 0.111200 | 0.097400 | 0.070800 |
| | .05 | 0.077200 | 0.056400 | 0.046400 | 0.027600 |
| | .01 | 0.018000 | 0.012200 | 0.009400 | 0.002600 |
| 20 | .20 | 0.293000 | 0.229800 | 0.226400 | 0.236600 |
| | .15 | 0.233600 | 0.180200 | 0.173200 | 0.190600 |
| | .10 | 0.173000 | 0.130600 | 0.116000 | 0.132200 |
| | .05 | 0.103600 | 0.072400 | 0.061600 | 0.065200 |
| | .01 | 0.029600 | 0.020600 | 0.014000 | 0.008000 |
| 25 | .20 | 0.290200 | 0.220400 | 0.197600 | 0.228000 |
| | .15 | 0.231800 | 0.170600 | 0.148200 | 0.173600 |
| | .10 | 0.170800 | 0.118200 | 0.099200 | 0.117000 |
| | .05 | 0.093200 | 0.063000 | 0.046400 | 0.056400 |
| | .01 | 0.022000 | 0.012000 | 0.006800 | 0.006000 |
| 30 | .20 | 0.195400 | 0.668200 | 0.652400 | 0.288600 |
| | .15 | 0.150400 | 0.605600 | 0.588000 | 0.222800 |
| | .10 | 0.097000 | 0.525400 | 0.503000 | 0.147400 |
| | .05 | 0.050000 | 0.406200 | 0.373600 | 0.071400 |
| | .01 | 0.012400 | 0.187200 | 0.151200 | 0.012000 |
| 40 | .20 | 0.310800 | 0.228800 | 0.214200 | 0.363600 |
| | .15 | 0.246200 | 0.174000 | 0.161400 | 0.280600 |
| | .10 | 0.186400 | 0.120600 | 0.112400 | 0.210600 |
| | .05 | 0.106200 | 0.066400 | 0.056600 | 0.099400 |
| | .01 | 0.031000 | 0.018200 | 0.012200 | 0.020000 |
| 50 | .20 | 0.321800 | 0.236800 | 0.230000 | 0.462400 |
| | .15 | 0.263400 | 0.183800 | 0.176400 | 0.386000 |
| | .10 | 0.198000 | 0.130200 | 0.120200 | 0.284800 |
| | .05 | 0.122200 | 0.078000 | 0.073200 | 0.145400 |
| | .01 | 0.038600 | 0.014800 | 0.014200 | 0.037000 |

Table 4.12. In this table, MLE and MDE and W statistic are used

# V. Conclusions and Recommendations

## 5.1 Conclusions

In this thesis, the Anderson-Darling and Modified W statistic critical values for the three parameter Weibull distribution when all three must be estimated from the sample first estimating parameters by MLE, later keeping the scale and shape parameters constant reestimate the location parameter, then recalculate the other to parameters by MLE , are valid. In the power study, a true null hypothesis achieved the expected level of significance. Also, another power study conducted for three different values of the each parameter. In this study, a true null hypothesis achieved in all of the 27 different parameter sets. From the latter experiment I concluded that all the study can be made for only one set of parameters because the Extreme Value Distribution only location and scale parameters ( this is proven by this experiment having achieved the true null hypothesis in each set of parameters ). The conclusions can be summarized as follows:

- Test based on A–D statistic

  1. The Weibull distribution was able to fit the Normal, and Gamma data. This shows the importance of the location parameter of the Weibull Distribution.
  2. The power was very good when the alternative distribution was Uniform. My conclusion from this result is the Uniform data can not be fitted by the Weibull distribution because even though the Weibull can take many shapes it can not take a shape close to the Uniform distribution.
  3. The power was also high when the alternative distribution was Beta.
  4. As the sample size increased

     * the power increased for the Uniform and Beta distributions.

* the power was stable for the Normal and Gamma distributions at a given significance level but very slightly increased.

– Test based on $\tilde{W}^*$ statistic

* The power for the alternate Uniform and Beta distributions was higher than the A–D statistics power.
* In using this statistic the Gamma distribution's power was significant.
* Given the alternative distribution, the test can be made as only one-tailed test. For example there was nearly no rejection on the upper tail when the alternative distribution was the Uniform distribution.
* But for the Normal distribution this statistic also says that the normal data can be fitted by Weibull distribution.

General conclusions

1. This is the first test of the three parameter Weibull distribution where the significance level of the test and the invariance of the test has been examined.
2. This test has the additional property of demonstrating the robustness of the three parameter Weibull distribution for modelling Normal,Beta and Gamma distributions. So, we can say that the Weibull distribution may provide enough flexibility neede to make a model sufficiently accurate for use in modelling or in an analysis.
3. The three parameter Weibull Distribution has great potential as an alternative model for the normal distribution. Because both tests showed that the normal distribution had achieved given significance level.

   The $\tilde{W}^*$ statistic appears to be more powerful than A–D statistic. Especially, when the Alternative distribution is known, one-tailed test can be made. This will increase this statistic's power. Because in Uniform alternatives, there was no rejection in the lower tail, and there was some significance in upper tail–lower tail rejection numbers and pattern.

4. the two statistic are very different from each other, one is based on minimizing the distance between the EDF and CDF, the other one is based on the comparison of the two different scale estimates. A–D test showed that Gamma, and Normal distributions can be fitted reasonably well by the Weibull Distribution while the W test contradicts the Gamma Distributions result.

   In this research, two test statistics are used. In the literature there is no goodness of fit test for the three parameter Weibull distribution. Only one critical value table for each statistic is obtained. A statistican can test its data by using only one table chosen a test statistic ( from A–D or W statistic ). We hope this research will be a step towards finding better test statistics for the three parameter Weibull distribution and increase the popularity of the Weibull distribution. The conclusions based on the power study presen.ed in Chapter IV are applicable to the 8 alternate distributions.

### 5.2 Recommendations for the Further Research

The following recommendations can be investigated in the future.

* In the three parameter case, in order to reach to the asymptotic points Monte Carlo Simulation study should be extended to 20000 repetitions.
* A comparison can be made with the Chi-Square test against the proposed tests.
* The $\tilde{W}^*$ statistic is a two-tailed test statistic. This statistic should be used as a one tailed test statistic against an alternative distribu-

tion known. This will increase the power of the test against known alternative.

* Some authors criticized the Harter and Moore's method since in this method we are really looking for a local maximum instead of Global maximum. But As stated in Chapter IV and defended by Smith [26:360] one can make the likelihood function infinite when the location estimate approaches to the first order statistic. This problem should be addressed in the future reseachs.
* Other invariant estimation techniques should be tried for the two parameter extreme value distribution left after the transformation.

## *Bibliography*


1. Banks, Jerry and John S. Carson. *Discrete-Event System Simulation.* Englewood Cliffs:Prentice-Hall,1984.
2. Cheng,R.C.H.,and N.A.K. AMIN." Estimating parameters in continuous Univariate Distributions with a shifted origin," *J.Roy.Statist. Soc. B,45:* 394-403 (1983).
3. Cohen, A.C. and B.J.Whitten. *Parameter Estimation in Reliability and Life Span Models.* New Yorkand Basel:Marcel Dekker Inc., 1988.
4. Crown, J.S. *A Goodness-of-Fit Test For The Three Parameter Weibull Using Minimum Distance Estimation of Parameters.* MS Thesis AFIT, School of Engineering, Air Force Institute of Technology (AU), WPAFB OH, March 1991.
5. D'Agostino,R.B."Linear estimation of the Weibull parameters," *Technometrics,13:* 171-182 (1971).
6. Devore, J.L. *Probability and Statistics for Engineering and the Sciences.* Pacific Grove:Brooks-Cole Publishing Company, 1991.
7. Dubey,S.D. "On some permissible estimators of the location parameter of Weibull and certain other distributions," *Technometrics,11:* 683-690 (1969).
8. Gallagher,M.A.. *Robust Statistical Estimation through Minimum Distance Using the Three-parameter Weibull.* MS Thesis AFIT, School of Engineering, Air Force Institute of Technology (AU), WPAFB OH, DECEMBER 1986.
9. Harter,H.L. and Moore,A.H." Maximum likelihood estimation, from doubly cencored samples, of the parameters of the first asymptotic distribution of extreme values," *Journal of American Statistical Association, 63:* 889-901 (1968).
10. Harter, H.L. and A.H.Moore. "Maximum Likelihood Estimation of the Parameters of Gamma and Weibull Populations from Complete and from Censored Samples," *Technometrics 7:* 639-643 (1984).
11. Hines,William W. and DouglaS C.Montgomery. *Probability and Statistics in Engineering and Management Science.* New York:John Wiley & Sons Inc., 1980.
12. Hirose,Hideo."Percentile point estimation in the three parameter Weibull distribution by the extended maximum likelihood estimate," *Computational Statistics & and Data Analysis 11:* 309-331 (1991).
13. Ireson,W.G. *Reliability handbook.* New York:Mc Graw-Hill Inc.,1966.

14. Johnson,N.L. and Samuel Kotz. *Continuous univariate distributions -1.* New York:John Wiley & Sons Inc., 1970.

15. Kapur, K.C., L.R.Lamberson. *Reliability in Engineering Design.* New York:John Wiley & Sons Inc., 1977.

16. Law, A.M., W.D.Kelton. *Simulation Modeling and Analysis.* New York:McGraw-Hill Inc., 1991.

17. Lehmer,D.H."Mathematical methods in large-scale computing units," *Ann. Comput. Lab. Harvard Univ., 26:* 141-146 (1951).

18. Mann,N.R.,Scheuer,E.M. and Fertig,K.W."A new goodness of fit test for the Weibull distribution or extreme value distribution with unknown parameters," *Communications in Statistics, 2:* 383-400 (1973).

19. Miller,R.*Robust Minimum Distance Estimation of the three parameter Weibull.* MS Thesis AFIT, School of Engineering, Air Force Institute of Technology (AU), WPAFB OH, DECEMBER 1980.

20. Ozturk,A.,S.Korukoglu." A new test for the extreme value distribution," *Commun. Statist. -Simula.,17(4):* 1375-1393 (1988).

21. Parr, W.C. "Minimum Distance and Robust Estimation,". PH.D. Thesis, Southern Methodist University, 1978.

22. Pritsker, A. Alan B.'*Introduction to Simulation and SLAM II.* New York:John Wiley & Sons Inc.,1984.

23. Sahler, W."Estimation by Minimum-Discrepancy Methods," *Metrica 16:* 85-106 (1970).

24. Shapiro,S.S. and Brain,C.W." W−test for the Weibull distribution," *Comm. in Statistics. Simulation and Computation,16(1):* 209-219 (1987).

25. Stephens, M.A. and R.D.D'Agostino. *Goodness of Fit Techniques.* Marcel Dekker Inc. 1986.

26. Smith,R.L. and Naylor,J.C." A comparison of Maximum Likelihood and Bayesian Estimators for the three parameter Weibull distribution," *Applied statistics, 36(3):* 358-369 (1987).

27. von Neumann,J."Various techniques used in connection with random digits," *Natl. Bur. Std. Appl. Math. Ser.,12:* 36-38 (1951)

28. Wolfowitz, J. "The Minimum Distance Method," *Annals of Mathematical Statistics 28:* 75-88 (1957).

29. Zanakis,S.H."A simulation study of some simple estimators for the three-parameter Weibull Distribution," J.Comput.Simul. 9: 101-106 (1979).

# Appendix A. Computer Program

```
program thesis(a1,a2.out,input,output);
   const
      error     = 0.000001;   (* error and tolerance are limits *)
      tolerance = 0.000001;   (* used in the numerical routines *)
      repetitions = 5000;
   type
      gofstat  =(AD,CVM,ADEXT);
      teststat=array[0..5001] of real;
      critv=array[0..5] of real;
      critW=array[0..10] of real;
      critvalues=array[1..13,1..10] of real;
      wmod=array [1..1001] of real; (* generated random variables       *)
      data = array[0..40] of real;  (* position 0 is the number of rvs. *)
      para = array[1..3] of real;   (* array of the Weibull parameters  *)
      logpara=array [1..2] of real; (* in order location, scale, shape  *)
   var
      NRrejAD,NRrejCVM,NRrejWup,NRrejWlow,cvad,cvcvm:critv;
      cvW:critW;
     ADCRIT,CVMCRIT,wmodstat:teststat;
     logdataset:data;
     logmme:logpara;
     inputs,a1,a2,out:text;(* out is the output file                       *)
     z,  FOSL,             (* cumulative value at each data point           *)
     dataset:data;         (* generated random numbers                      *)
     TRU,                  (* True parameters used to generate the data     *)
     MLE,                  (* Maximum Likelihood Estimates                  *)
     MDLAD,                (* Min Dist on Location using Anderson-Darling   *)
     MDLCVM : para;        (* Min Dist on Location using Cramer-Von Mises   *)
     which   : gofstat;    (* Which goodness of fit statistic AD or CVM     *)
     mlemodcvm, temp,      (* modified CVM for MLE parameters               *)
     mleerror,             (* mle error for a particular parameter          *)
     mlegof,power,         (* mle goodness of fit statistic                 *)
     gofvalue,wmodst: real;(* each estimates goodness of fit statistic      *)
     seed,mlefails,        (* seed for uniform random number generator      *)
     i,number,n,k,j ,nn,num: integer;
     mlefailed,            (* MLE procedure failed to converge              *)
     trueloc : boolean;    (* true if location is assumed known for MLE     *)
     value : array[1..5] of real ;
```

```
function uniform(var seed:integer): real;
(*  Generates a uniform random number *)
(*  Introduction to Simulation by Payne(1982) page 310 *)
   const
      a = 16807;
      c = 0.0;
      m = 2147483647;

   var
      temp : real;
   begin
      temp := (a/m) * seed;
      temp := temp - trunc(temp);
      seed := trunc(m*temp);

      if seed = 0 then
 seed := 1;

      uniform := seed / m;
   end; (* function uniform *)

function gamma(m:integer) : real;
   var
      i : integer;
      temp,temp2 : real;
   begin
      temp2 :=1.0;
      for i := 1 to m do
         begin
            temp   := abs(uniform(seed));
            temp2  := temp2 * temp;
         end;
      gamma := (-1/m)*ln(temp2);
   end;

function cvmgof(x:data;param:para): real;
(*  This function returns the Cramer Von-Mises Goodness of Fit     *)
(*  Statitic for the three parameter Weibull.                      *)
(*  Formulas published in Woodruff, Moore, and Dunne (1983)        *)
(*  Data must be ORDERED !                                         *)
   var
      cum   : data;            (* cummulative distribution *)
```

```
      i,num : integer;
      sum,temp  : real;
      begin
      num := trunc(x[0]);
      for i := 1 to num do
 begin
    if x[i] <= param[1] then  cum[i] := 0 else
 begin
     temp := -1*exp(param[3] * ln((x[i]-param[1])/param[2]));
     if temp < -20 then cum[i] := 1.0  else    cum[i] := 1 - exp(temp);
  end;
 end; (* for *)
      sum := 0;
      for i := 1 to num do
 begin
    temp := cum[i] - (2*i - 1)/(2*num);
    sum := sum + temp*temp;
 end; (* for *)
      cvmgof := (1/(12*num)) + sum;
   end; (* function cvmgof *)

function adgof(x:data;param:para): real;
(*  This function returns the Anderson-Darling Goodness of Fit     *)
(*  Statitic for the three parameter Weibull.                      *)
(*  Formulas published in Woodruff, Moore, and Dunne (1983)        *)
(*  Data must be ordered!                                          *)
   var
      cum : data;          (* cummulative distribution                    *
      i,num : integer;     (* num is number of data values                *
      sum, temp : real;
begin
      num := trunc(x[0]);
      for i := 1 to num do
 begin
    if x[i] <= param[1] then  cum[i] := 0.000001   else
      begin
 temp := -1*exp(param[3] * ln((x[i]-param[1])/param[2]));
 cum[i] := 1 - exp(temp);
 if cum[i] < 0.001 then cum[i] := 0.001;
 if cum[i] > 0.999 then cum[i] := 0.999;
      end;
 end; (* for *)
```

```
      sum := 0;
      for i := 1 to num do
 begin
    temp := (2*i -1)*(ln(cum[i]) + ln(1-cum[num+1-i]));
    sum := sum + temp;
 end; (* for *)
      adgof := -1*num - (sum/num);
   end; (* function adgof *)
function cvmextgof(x:data;param:para): real;
(*  This function returns the Cramer Von-Mises Goodness of Fit     *)
(*  Statitic for the three parameter Weibull.                      *)
(*  Formulas published in Woodruff, Moore, and Dunne (1983)        *)
(*  Data must be ORDERED !                                         *)
   var
      cum   : data;            (* cummulative distribution *)
      i,num : integer;
      sum,temp  : real;
      begin
      num := trunc(x[0]);
      for i := 1 to num do
 begin
         temp := -exp(-(x[i]-param[1])/param[2]);
  cum[i] := exp(temp);
  if cum[i] < 0.001 then cum[i] := 0.001;
  if cum[i] > 0.999 then cum[i] := 0.999;
  end; (* for *)

      sum := 0;
      for i := 1 to num do
 begin
    temp := cum[i] - (2*i - 1)/(2*num);
    sum := sum + temp*temp;
 end; (* for *)
      cvmextgof := (1/(12*num)) + sum;
   end; (* function cvmextgof *)


function adextgof(x:data;param:para): real;
(*  This function returns the Anderson-Darling Goodness of Fit     *)
(*  Statitic for the three parameter Weibull.                      *)
(*  Formulas published in Woodruff, Moore, and Dunne (1983)        *)
(*  Data must be ordered!                                          *)
```

```
    var
        cum : data;          (* cummulative distribution                          *)
        i,num : integer;     (* num is number of data values                      *)
        sum, temp : real;
 begin
        num := trunc(x[0]);
        for i := 1 to num do
  begin
           temp := -exp(-(x[i]-param[1])/param[2]);
   cum[i] := exp(temp);
   if cum[i] < 0.001 then cum[i] := 0.001;
   if cum[i] > 0.999 then cum[i] := 0.999;
   end; (* for *)
       sum := 0;
       for i := 1 to num do
  begin
     temp := (2*i -1)*(ln(cum[i]) + ln(1-cum[num+1-i]));
     sum := sum + temp;
  end; (* for *)
       adextgof := -1*num - (sum/num);

   end; (* function adextgof *)

function gof(x:data; pars:para; which:gofstat):real;
(* selects which "goodness of fit " statistic to evaluate *)
   begin
      if which = AD then    (* Anderson-Darling Goodness of Fit Statistic *)
 gof := adgof(x, pars)
      else if which = CVM then (*Cramer-von Mises Goodness of fit statistic*)
 gof := cvmgof(x, pars);
      end;

function cumweibull(dist:para;x:real): real;
       (* returns the cummulative weibull value for point x   *)
       (* dist contains the weibull location, scale and shape *)
  var
     temp : real;
  begin (* of function cumweibull *)
     if x <= dist[1] then
cumweibull := 0.0
     else
begin
```

```
   temp := exp(dist[3]*ln((x-dist[1])/dist[2]));
   if temp > 20 then
      cumweibull := 1.0
   else
      cumweibull := 1 - exp(-1*temp);
end;
  end; (* of function cumweibull *)

procedure MLEest(x:data;ctrue:boolean;var location,scale,shape:real;
var mlefailure:boolean);
   (***************************************************************)
   (*  Maximum-Likelihood estimation of three parameters weibull  *)
   (*  Iterative technique developed by H. Leon Harter and Albert *)
   (*  H. Moore and published in Technometrics (Nov 1965).        *)
   (*  Formulas (for two parameter) from ATC Notes page 235 were  *)
   (*  adjusted.  Also see Miller's 1980 thesis.                  *)
   (*  When location is unknown this procedure occasionally fails *)
   (*  to converge                                                *)
   (***************************************************************)
   var
      r,                      (* number of data points less than location    *)
      iterations,             (* number of times both location and shape est *)
      num    : integer;       (* number of data points                       *)
      a,b,
      k,                      (* another name for shape used in locationest  *)
      mletol,                 (* error tolerance for mle procedure           *)
      lastshape,
      lastlocation : real;
  function shapeest(c,betaorg:real):real;  (* part of MLE procedure *)
  (* shapeest estimates the shape parameter for a given location *)
      var
beta1,            (* betaorg is original shape   *)
beta2,            (* iterative estimate of shape *)
sumlnx,           (* sum of ln(x[i])             *)
temp,
temp2 : real;
i     : integer;
     begin
beta1 := betaorg;
repeat            (* iterative loop until beta converges *)
   begin
      temp := 0;
```

```
      temp2 := 0;
      sumlnx := 0;
      for i := (r+1) to num do  (* formula from ATC notes *)
  begin
          temp := temp + exp(beta1*ln(x[i]-c))*ln(x[i]-c);
     temp2 := temp2 + exp(beta1*ln(x[i]-c));
     sumlnx := sumlnx + ln(x[i] - c);
  end;
      beta2 := (num-r)/(((num-r)*temp/temp2) - sumlnx);
      beta1 := (2*beta2 + beta1)/3;  (* formula overshoots *)
      if (beta1 > 91 ) and (betaorg = 91 ) then
 begin
  beta1 :=91;
  beta2 :=91;
 end;
      if beta1 > 91 then (* prevents unstable overshoots *)
 begin
  beta1 :=91;
  betaorg :=91;
 end;
   end;
 until abs(beta1 - beta2) < mletol,
 k := beta1;
 shapeest := beta1;
     end; (* function shapeest *)
  function scaleest:real;  (* part of MLE procedure  *)
  (* scaleest estimatate the scale parameter.  The scale is determined by   *
  (* the location and shape paramters.  Formula in Miller's thesis. (1980)  *
     var
 temp : real;
 i    : integer;
     begin
 temp := 0;
 for i := (r+1) to num do
   temp := temp + exp(ln(x[i]-location)*shape);
 scaleest := exp(ln(temp/(num-r))/shape);
     end; (* function scaleest *)
     function partial(c:real):real;(* part of locationest under MLE proc. *)
     (* returns the derivative by c of ln(max likelihood function)     *)
     (* With the correct estimate of c the equation will be zero.      *)
     (* Harter and Moore (1965) found that with shape <= 1 the partial *)
     (* is monotone.  The resulting estimate is either 0 or the first  *)
```

```
    (* order statistic.  The function is positive with too low a c    *)
    (* and negative with too high a c.                                *)
   var
      sumx,         (* sum of 1/(x[i]-c)  (k is shape)  *)
      sumxck,       (* sum of (x[i]-c) to the kth power *)
      sumxck1:real; (* sum of (x[i]-c) to the k-1 power *)
   begin
      sumx := 0;
      sumxck := 0;
      sumxck1 := 0;
      i := 1;
      r := 0;
      while x[i] - c <= 0.0 do    (* censors data from below *)
begin
                      (* assumes data is ordered *)
   r := r + 1;
   i := i + 1;
end;
      for i := (r+1) to num do
 begin
    sumx := sumx + 1/(x[i]-c);
    sumxck := sumxck + exp(k*ln(x[i]-c));
    sumxck1 := sumxck1 + exp((k-1)*ln(x[i]-c));
 end;

      partial := (1-k)*sumx+(num*k*(sumxck1/sumxck));
   end; (* function partial *)
  function locationest(c:real):real;  (* part of MLE procedure *)
  (* locationest estimates the location.  The iterative technique was used *)
  (* by Harter and Moore (1965).  Their equation 3.5 is simplified in that *)
  (* only complete samples are allowed.  Values for location are tried and *)
  (* then adjusted to until the equation is equal to zero.                 *)
     var
upper,          (* upper limit on c               *)
lower,          (* lower limit on c               *)
value,          (* value of partial derivative of L *)
lowerval,       (* value at lower limit           *)
upperval :real; (* value at upper limit           *)
i  : integer;   (* c is estimate of location      *)
     begin (* function locationest *)
value := partial(c);  (* c is last estimate of location           *)
```

```
 if value > 0 then      (* bound c between a lower and higher value *)
    begin               (* the lower value is pos and higher is neg *)
       lower := c;
       lowerval := value;
       if (c + 1.0) < x[i] then  (* try to get a small interval *)
  begin
     upper := c + 1.0;
     upperval := partial(upper);
     if upperval > 0 then
begin
   upper := x[1];
   upperval := partial(upper);
end;
  end
else (* c+1 > x[1] (too close to end of interval )*)
  begin
     upper := x[1];
     upperval := partial(upper);
  end;
    end (* if value > 0 *)
 else if value < 0 then
    begin
       upper := c;
       upperval := value;
     lower := c - 1.0;
     lowerval := partial(lower);
   while (lowerval < 0.0) do
                           begin
                            lower:=lower-1.0;
                            lowerval:=partial(lower);
                             if (lowerval > 0 ) then
                              upper:=lower+1.0;
                              upperval:=partial(upper);
          end;

    end
else if abs(value) < mletol then    (* if is zero then quit *)
   begin
      upper := c;
      lower := upper;     (* prevents entering loop below *)
   end;
if abs(upperval) < mletol then
```

```
      begin
         c := upper;
         lower := upper;
      end;
   if abs(lowerval) < mletol then
      begin
         c := lower;
         upper := lower;
      end;
   while ((upper-lower)>mletol) and (abs(value) > mletol) do
      begin
         c := (upper + lower)/2;   (* binary search for zero *)
         value := partial(c);
         if value > 0 then
    begin
       lower := c;
       lowerval := value;
    end;
         if value < 0 then
    begin
       upper := c;
       upperval := value;
    end;
      end; (* while *)
   i := 1;
   r := 0;
   while x[i] <= c do    (* censors data from below *)
      begin              (* r is used in shapeest   *)
         r := r + 1;
         i := i + 1;
      end;
   locationest := c;
        end; (* function locationest *)
    begin (* procedure MLEest *)
       mlefailure := false;
       num  := trunc(x[0]); (* the number of data points         *)
       if ctrue  then (* location is known *)
 begin
    i := 1;
    r := 0;
    while x[i] <= location do    (* censors data from below *)
       begin
```

```
r := r + 1;
i := i + 1;
    end;
  shape := 3.0;
  mletol := tolerance;
  shape := shapeest(location,shape);
end
    else (* location is unkown *)
begin
   mletol := tolerance;
   shape :=1.0;
   b := 0.9*x[1];
   r := 0;
          a:=-20;
   shape := shapeest(b,shape);
   location := locationest(b);
     iterations := 1;
      begin
iterations := 1;
lastlocation := 0;
lastshape := 0;

     while (abs(shape-lastshape)> mletol) and
           (abs(location-lastlocation) > mletol) and
  (iterations <= 1050) and (shape <90) do
       begin
mletol := 100*tolerance;
while ((abs(shape-lastshape) +
              abs(location-lastlocation)) > mletol) and
  (iterations <= 1050) and (shape <90) do
begin
   lastshape := shape;
   shape := shapeest(location,shape);
   lastlocation := location;
   location := locationest(location);
 if location > x[1] then
    begin
writeln('location buyuk x1 den',location);
      r :=1;
      i :=1;
      while x[i] < x[1] do
begin
```

```
  r:=r+1;
  i:=i+1;
end;
     end;
    if r >=1 then (* prevents cycling by not reestimating shape *)
begin     (* location after shape when cencoring occurs *)
   lastlocation := location; (* ends loop *)
   shape := shapeest(location,shape);
   lastshape := shape;
end;
     iterations := iterations + 1;
     end; (* 2nd while *)
       (* increase tolerance *)
     end; (* 1st while *)
          if shape >90 then
begin
    mlefailure := true;
    writeln('MLE shape too large in',iterations:3,'iterations.');
end
      else
     mlefailure := false;
     end;
      end;
     scale := scaleest;
 a := partial(location);
if (abs(a) > 0.0001)  and (not trueloc) then
 mlefailure := true;


  end; (* of procedure MLEest *)

procedure findcrit(x:teststat; var cvpass:critv);
   var
      mr    : teststat;
      m,b,alpha : real;
      i,num: integer;
   function cv(x,mr:teststat;alpha:real):real;
      var
         m,b : real;
         i : integer;
      begin
         for i := 0 to num do
```

```
              begin
                 if (mr[i] < alpha) and (mr[i+1]>alpha) then
                    begin
                       m :=(mr[i+1]-mr[i])/(x[i+1]-x[i]);
                       b := mr[i] - m*x[i];
                       cv := ( alpha - b )/m;
                    end
                 else if mr[i] = alpha then
                    cv := x[i];
             end;
        end;
    begin
        num := repetitions;
        for i:= 1 to num do
           begin
              mr[i] := (i-0.3)/(num+0.4);
           end;
        mr[0] := 0 ;
        mr[num+1] := 1;
        m := (mr[2] -mr[1])/(x[2]-x[1]);
        b := mr[1] - m*x[1];
        x[0] := -b/m;
        if x[0] < 0 then
           x[0] := 0;
        m := (mr[num]-mr[num-1])/(x[num]-x[num-1]);
        b := mr[num] - m*x[num];
        x[num+1] := (1.0 - b)/m;
        writeln('alpha critical value');
        for i := 1 to 5 do
           begin
              alpha := 0.75 + 0.05*i;
              if alpha > 0.96 then
                 alpha := 0.99;
              cvpass[i] := cv(x,mr,alpha);
              writeln(alpha:8:4,cvpass[i]:10:6);
           end;
     end;

  procedure findcritW(x:teststat; var cvpass:critW);
     var
        mr    : teststat;
        m,b,alpha : real;
```

```
      i,num: integer;
   function cv(x,mr:teststat;alpha:real):real;
      var
         m,b : real;
         i : integer;
      begin
         for i := 0 to num do
            begin
               if (mr[i] < alpha) and (mr[i+1]>alpha) then
                  begin
                     m :=(mr[i+1]-mr[i])/(x[i+1]-x[i]);
                     b := mr[i] - m*x[i];
                     cv := ( alpha - b )/m;
                  end
               else if mr[i] = alpha then
                  cv := x[i];
            end;
      end;
begin
      num := repetitions;
      for i:= 1 to num do
         begin
            mr[i] := (i-0.3)/(num+0.4);
         end;
      mr[0] := 0 ;
      mr[num+1] := 1;
      m := (mr[2] -mr[1])/(x[2]-x[1]);
      b := mr[1] - m*x[1];
      x[0] := -b/m;
      if x[0] < 0 then
         x[0] := 0;
      m := (mr[num]-mr[num-1])/(x[num]-x[num-1]);
      b := mr[num] - m*x[num];
      x[num+1] := (1.0 - b)/m;
      writeln('alpha critical value for upper');
      for i := 1 to 5 do
         begin
            alpha := 0.875 + 0.025*i;
            if alpha > 0.975 then
               alpha := 0.995;
            cvpass[i] := cv(x,mr,alpha);
            writeln(alpha:8:4,cvpass[i]:10:6);
```

```
         end;
      writeln('alpha critical value for lower');

       for i := 6 to 10 do
         begin
            alpha := 0.125 - 0.025*(i-5);
            if alpha=0 then
               alpha := 0.005;
            cvpass[i] := cv(x,mr,alpha);
            writeln(alpha:8:4,cvpass[i]:10:6);
         end;

   end;




procedure GoldenSearch(x:data; which:gofstat; var pars:para);
   (* starting at "a" searches in "direction" until the function stops*)
   (* decreasing.  Then begins a golden search on the last two        *)
   (* intervals just prior to the function increasing.               *)
   (* location should have bounded below the first order statistic    *)
   var
      a,b,                 (* current right and left endpoints   *)
      ab,                  (* midpoint between a and b           *)
      left,right,          (* golden search midpoints            *)
      fa,fab,fb,
      fleft,fright,        (* function value at current points   *)
      step,                (* line search interval lenth         *)
      r,                   (* sets golden search interval width  *)
      bound: real;         (* golden search iteration error bound*)

   begin
      step := pars[1]/20;              (* line interval step size      *)
      r := 0.618034;                   (* golden search multiplier     *)

      a := pars[1];
      fa := gof(x,pars,which);         (* current objective value      *)
      fb := fa + 1;                    (* initiate loop                *)

      while (fb - fa) > error do       (* loop determines direction to *)
begin                          (* decrease the function or if  *)
```

```
    b := a + step;           (* current point is the minimum *)
    pars[1] := b;
    fb := gof(x,pars,which);

    if fb > fa then           (* try the other direction   *)
       begin
  step := -1 * step;
  b := a + step;
  pars[1] := b;
  fb := gof(x,pars,which);
       end;
    step := step/4;
 end;

      if fb > fa then           (* the original point was the minimum *)
    pars[1] := a
      else
 begin                    (* line search to find interval with minimum *)
    ab  := a;                  (* initialize search  *)
    fab := fa;

    repeat                (* line search checks every step to find     *)
       a   := ab;         (* where the function starts to increase     *)
       fa  := fab;
       ab  := b;
       fab := fb;
       b   := b + step;
       pars[1] := b;
       fb  := gof(x,pars,which);
    until (fb > fab);

    left  := b - r*(b-a);    (*** GOLDEN SEARCH begins ***)
    right := a + r*(b-a);
    bound := 2 * abs(step);

    pars[1] := left;
    fleft := gof(x,pars,which);
    pars[1] := right;
    fright:= gof(x,pars,which);

    while abs(fb-fa) > error do
       begin
```

```
   if fleft < fright then          (* delete right interval *)
      begin
  b := right;
  fb := fright;
  right := left;
  fright := fleft;
  left := b - r*(b-a);
  pars[1] := left;
  fleft := gof(x,pars,which);
      end; (* if *)

   if fright <= fleft then         (* delete left interval *)
      begin
  a := left;
  fa := fleft;
  left := right;
  fleft := fright;
  right := a + r*(b-a);
  pars[1] := right;
  fright := gof(x,pars,which);
      end; (* if *)
   bound := r*bound;
        end; (* of while *)

    if fleft < fright then
       begin
   if fa < fleft then
      pars[1] := a
   else
      pars[1] := left
       end
    else
       begin
   if fb < fright then
      pars[1] := b
   else
      pars[1] := right;
       end;
end; (* of else (from long time ago) *)

     trueloc := true;
```

```
      MLEest(x,trueloc,pars[1],pars[2],pars[3],mlefailed);

      (* re estimate shape and scale using min dist estimate of location *)
    end; (* of procedure goldensearch *)
  procedure bubble(var critsort:teststat);
    var
      number,i,j:integer;
      temp : real;
    begin
      number := repetitions;
      for j := (number-1) downto 1 do
        for i:= 1 to j do
          begin
            if critsort[i] > critsort[i+1] then
              begin
                temp := critsort[i];
                critsort[i] := critsort[i+1];
                critsort[i+1]:= temp;
              end;
          end;
    end;
  procedure datasort(var dataset:data);
    var
      number,i,j:integer;
      temp:real;
    begin
      number := trunc(dataset[0]);
      for j :=(number-1) downto 1 do
        for i:= 1 to j do
          begin
            if dataset[i] > dataset[i+1] then
              begin
                temp := dataset[i];
                dataset[i] := dataset[i+1];
                dataset[i+1]:=temp;
              end;
          end;
    end;


  procedure Weibull(TR:para; var dataset:data);
```

```
(* Weibull generates a data set of Weibull random variables *)
(* Uses the inverse transform technique. Banks and Carson   *)
(* (1984) pages 294-300.                                     *)
(* DATA MUST BE ORDERED FOR OTHER PROCEDURES                 *)
   var
      number,
      i,j : integer;
      temp : real;
   begin
      number := trunc(dataset[0]);  (* number of random variables (<30) *)
      for i := 1 to number do
 begin
    temp := uniform(seed);
  dataset[i] := TR[2]*exp((1/TR[3])*ln(-ln(temp))) + TR[1];
 end;
     datasort(dataset);
        end; (* Weibull *)
procedure Normal(var dataset:data),
 (* mean= 15 and std dev= 2.*)
   var
      number,i,j:integer;
      temp2,temp:real;
   begin
      number := trunc(dataset[0]);
      j := trunc((number+1)/2);
      for i:= 1 to j do
         begin
            temp :=uniform(seed);
            temp2:=uniform(seed);
            temp :=sqrt(-2*ln(temp));
            temp2:=temp2*6.2831853;
            dataset[i] := 15 + 2 * temp*cos(temp2);
            dataset[i+j]:=15 + 2 * temp*sin(temp2);
         end;
       datasort(dataset);
   end;
procedure Normal1(var dataset:data);
 (* mean= 12 and std dev= 1.*)
   var
      number,i,j:integer;
      temp2,temp:real;
   begin
```

```
      number := trunc(dataset[0]);
      j := trunc((number+1)/2);
      for i:= 1 to j do
         begin
            temp :=uniform(seed);
            temp2:=uniform(seed);
            temp :=sqrt(-2*ln(temp));
            temp2:=temp2*6.2831853;
            dataset[i] := 12 +  temp*cos(temp2);
            dataset[i+j]:=12 +  temp*sin(temp2);
         end;
       datasort(dataset);
   end;

procedure Erlang(m:integer;var dataset:data);
   var
      number,i,j:integer;
      temp2,temp:real;
   begin
      number := trunc(dataset[0]);
      for i:=1 to number do
         begin
            temp2 := 1.0;
            for j:= 1 to m do
               begin
                  temp := uniform(seed);
                  temp2:= temp2*temp;
               end;
            dataset[i]:=(-m*0.25)*ln(temp2)+10.0;
         end;
      datasort(dataset);
   end;
procedure Beta(p,q:integer;var dataset:data);
   var
      i,number:integer;
      x1,x2   :real;
   begin
      number := trunc(dataset[0]);
      for i:=1 to number do
         begin
            x1 := gamma(p);
            x2 := gamma(q);
```

```
            dataset[i] := x1/(x1+x2) + 10.0;
         end;
      datasort(dataset);
   end;
procedure Unif(var dataset:data);
(* between 10 and 15*)
   var
      number,i:integer;
      temp    :real;
   begin
      number := trunc(dataset[0]);
      for i:= 1 to number do
         begin
            temp := uniform(seed);
            dataset[i] := 5.0* temp + 10.0;
         end;
      datasort(dataset);
   end;
procedure Unif1(var dataset:data);
(* between 8 and 12*)
   var
      number,i:integer;
      temp    :real;
   begin
      number := trunc(dataset[0]);
      for i:= 1 to number do
         begin
            temp := uniform(seed);
            dataset[i] := 4.0* temp + 8.0;
         end;
      datasort(dataset);
   end;

procedure Getdata(k:integer;var dataset:data);
   var
      TR :para;
   begin
      if k=1 then
         begin
            TR[1]:=10.;
            TR[2]:= 4;
            TR[3]:= 3.0;
```

```
      Weibull(TR,dataset);
   end
else if k=2 then
   begin
     Unif(dataset);
   end
else if k=3 then
   begin
      Unif1(dataset);
   end
else if k=4 then
   begin
      Erlang(3,dataset);
   end
else if k=5 then
   begin
      Erlang(4,dataset);
  end
else if k=6 then
   begin
      Erlang(5,dataset);
   end
else if k=7 then
   begin
      Normal(dataset);

   end
else if k=8 then
   begin
     Normal1(dataset);
   end
else if k=9 then
   begin
      Beta(2,2,dataset);

   end
else if k=10 then
   begin
      Beta(2,3,dataset);

   end
```

```
      end;

procedure logtrans(x:data;var wmodstat:real);
const
   euler=0.5772156649;
var
    n:integer;
    temp,botemp,wixitemp,wnixitemp,wisum,wnisum,w2ntemp,wntemp,
    witemp,wnitemp,sigmahat,b,wstar,mu,wixi,wnixi,temp1:real;
begin
    n:=trunc(x[0]);
    temp:=0.0;
    temp1:=0.0;
    botemp:=0.0;
    wixitemp:=0.0;
    wnixitemp:=0.0;
    wisum:=0.0;
    wnisum:=0.0;
for i:=1 to n do
  begin
    logdataset[i]:=ln(dataset[i]-MDLAD[1]);

    temp:=(((2*i)-n-1)*logdataset[i])+temp;

    botemp:=logdataset[i]+botemp;
    if (i=n) then
      begin
      w2ntemp:=0.4228*n-wnisum;
      wntemp:=n-wisum;
      wixi:=wixitemp+wntemp*logdataset[i];
      wnixi:=wnixitemp+w2ntemp*logdataset[i];
      end
     else
      begin
      witemp:=ln((n+1)/(n+1-i));
      wixitemp:=wixitemp+witemp*logdataset[i];
      wnitemp:=witemp*(1+ln(witemp))-1;
      wnixitemp:=wnixitemp+wnitemp*logdataset[i];
      wisum:=wisum+witemp;
      wnisum:=wnisum+wnitemp;
      end;
  end;
```

```
sigmahat:= temp/(0.693147*n*(n-1));
b:=(0.6079*wnixi-0.257*wixi)/n;
wstar:=b/sigmahat;

mu:=(botemp/n)+euler*sigmahat;
temp1:=(0.49/sqrt(n))-(0.36/n);
wmodstat:=(wstar-1.0-(0.13/sqrt(n))+(1.18/n))/temp1;
(*writeln('sigmahat',sigmahat,'wstar',wstar,'mu',mu,'b',
         b,'wmodstat',wmodstat);*)

end;(*precedure logtrans*)




begin (* thesis *)
 {rewrite(out);}


   {seed := 207982; s10sh1}
   seed := 568432;
   TRU[1]:=10.;
   TRU[2]:= 4;
   TRU[3]:= 3.0;
   MLE[3]:=TRU[3];

   dataset[0] :=15;
   num:=15;
    j:=1;
writeln('dataset[0]',dataset[0]:2:0,'tru[3]',TRU[3],'num',num,'seed',seed);
   while j <= repetitions do
      begin
        mlefailed:=false;
       Weibull(TRU,dataset);
         trueloc := false;
  MLEest(dataset,trueloc,MLE[1],MLE[2],MLE[3],mlefailed);
 if not mlefailed then
         begin
for i:=1 to num do
begin
writeln(a2,dataset[i]);
end;
```

```
writeln(ai,MLE[1],MLE[2],MLE[3]);}
if ((j mod 100)=0) then
writeln(' j ',j);

    (****************************************************************)
(***   CALCULATE MINIMUM DISTANCE ESTIMATES                        ***)
(****************************************************************)

        for i:=1 to 3 do
           begin
               MDLAD[i]:=MLE[i];

           end;
        which :=AD;
        GoldenSearch(dataset,which,MDLAD);
        logtrans(dataset,wmodst);
        wmodstat[j]:=wmodst;

                 for i:=1 to num do
           begin
         dataset[i]:=-ln(dataset[i]-MDLAD[1]);
           end;
        datasort(dataset);
        MDLAD[1]:=-ln(MDLAD[2]);
        MDLAD[2]:=1/MDLAD[3];

        MDLAD[3]:=0.0;
        gofvalue := adextgof(dataset,MDLAD);
        ADCRIT[j]:=gofvalue;

     j:=j+1;
    end;
 if mlefailed then
 begin
    mlefails := mlefails+1;
end;
end;

j:=1;
   bubble(ADCRIT);
   bubble(wmodstat);
   findcrit(ADCRIT,cvad);
```

```
  findcritW(wmodstat,cvW);
 writeln('mlefails ',mlefails);

{for i:=1 to repetitions do
begin
writeln('ad ',ADCRIT[i],i,'  ',dataset[i]);
end;}
(*power study*)
mlefails:=0;
   for k:= 1 to 10 do
      begin
writeln('k ',k);
         for i:= 1 to 5 do
            begin
               NRrejAD[i]:=0;
               NRrejWup[i]:=0;
               NRrejWlow[i]:=0;

            end;
      while j <= repetitions do
        begin
if ((j mod 100)=0) then
 writeln(j);
               mlefailed:=false;
          Getdata(k,dataset);

               trueloc:=false;
               MLEest(dataset,trueloc,MLE[1],MLE[2],MLE[3],mlefailed);

           if not mlefailed then
             begin




               for i:=1 to 3 do
                  begin
                     MDLAD[i]:=MLE[i];
                    {MDLCVM[i]:=MLE[i];}
                  end;
               which := AD;
               GoldenSearch(dataset,which,MDLAD);
```

```
logtrans(dataset,wmodst);
for i:=1 to 5 do
    begin
      if wmodst > cvW[i] then
        NRrejWup[i]:=NRrejWup[i]+1;
      end;

 for i:=1 to 5 do
    begin
      if wmodst < cvW[i+5] then
        NRrejWlow[i]:=NRrejWlow[i]+1;
      end;

  for i:=1 to num do
begin
dataset[i]:=-ln(dataset[i]-MDLAD[1]);
end;
datasort(dataset);


MDLAD[1]:=-ln(MDLAD[2]);
MDLAD[2]:=1/MDLAD[3];
MDLAD[3]:=0.0;
gofvalue := adextgof(dataset,MDLAD);
for i:= 1 to 5 do
begin
    if gofvalue > cvad[i] then
       NRrejAD[i]:=NRrejAD[i]+1;
    end;



j:=j+1;
end;
if mlefailed then
begin
mlefails:=mlefails+1;
end;
end;
j:=1;
 writeln('j,',j,mlefails);
mlefails:=0;
```

```
            for i:=1 to 5 do
              begin
                power := NRrejAD[i]/repetitions;
               writeln('ad i=',i,'power=',power:8:6);
              end;
                   for i:=1 to 5 do
              begin
                power := (NRrejWup[i]+NRrejWlow[i])/repetitions;
               writeln('W i=',i,'power=',power:8:6,'up ',NRrejWup[i]:3:6,
                       ' low ',NRrejWlow[i]:3:6);
              end;
        end;
    end.
```

## *Vita*

1LT Erol YÜCEL was born on 1965 in Kütahya TURKEY. He graduated from the Isiklar Military High School in 1983 and entered the Turkish Air Force Academy. 1LT Yücel graduated from the Academy as a Second Lieutenant on 30 August 1987.

After graduating from the Air Logistic School in 1988, he was assigned to Balikesir AFB as a Stock Control Officer.

1LT Yücel worked for three years and was selected for the Postgraduate Education Program. He entered the School of Engineering, Air Force Institute of Technology, WPAFB, OH in 1991.

Permanent address: 2088 Sok. No:10-1
Çay Mah. Bayrakli
İZMİR - TURKEY

# REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

| | | |
|---|---|---|
| 1. AGENCY USE ONLY (Leave blank) | 2. REPORT DATE<br>March 1993 | 3. REPORT TYPE AND DATES COVERED<br>Master's Thesis |
| 4. TITLE AND SUBTITLE<br>MODIFIED GOODNESS-OF-FIT TESTS FOR THE WEIBULL DISTRIBUTION | | 5. FUNDING NUMBERS |
| 6. AUTHOR(S)<br>Erol Yücel, 1Lt, TUAF | | |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br>Air Force Institute of Technology, WPAFB OH 45433-6583 | | 8. PERFORMING ORGANIZATION REPORT NUMBER<br>AFIT/GOR/ENS/93M-25 |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | | 10. SPONSORING/MONITORING AGENCY REPORT NUMBER |
| 11. SUPPLEMENTARY NOTES | | |
| 12a. DISTRIBUTION/AVAILABILITY STATEMENT<br>Approved for public release; distribution unlimited | | 12b. DISTRIBUTION CODE |

13. ABSTRACT (Maximum 200 words)

This research will produce a new modified Anderson–Darling and $\tilde{W}^*$ Goodness of fit statistics for the three parameter Weibull Distribution when all parameters unknown and estimated by Maximum Likelihood-Minimum Distance combination. The critical values for each statistic, sample size and $\alpha$ levels 0.20,0.15,0.10,0.05 and 0.01 are produced. The Monte Carlo Simulation used 5000 repetitons for sample sizes of 10,15,20,25,30,40, and 50 with the Weibull shape equals to 3, scale equals to 4 and location equals to 10. The power study is made for the same sizes as above with the hypothesized Weibull Distribution against 8 other distributions.

| | | | |
|---|---|---|---|
| 14. SUBJECT TERMS<br>Weibull Distribution; Goodness-of-Fit; Monte Carlo Simulation; Minimum Distance; Maximum Likelihood | | | 15. NUMBER OF PAGES<br>89 |
| | | | 16. PRICE CODE |
| 17. SECURITY CLASSIFICATION OF REPORT<br>Unclassified | 18. SECURITY CLASSIFICATION OF THIS PAGE<br>Unclassified | 19. SECURITY CLASSIFICATION OF ABSTRACT<br>Unclassified | 20. LIMITATION OF ABSTRACT<br>UL |